A deep learning super-resolution model to speed up computations of coastal sea states

Jannik Kuehn, S. Abadie, B. Liquet, V. Roeber

Highlights

A deep learning super-resolution model to speed up computations of coastal sea states

J. Kuehn,S. Abadie,B. Liquet,V. Roeber

- Super-resolution substitutes high-resolution results from low-resolution computations
- The method reduces computation time by a factor of 50 while retaining good accuracy
- Super-resolution provides a more robust alternative to surrogate models
- Targeted data augmentation mitigates large errors in extreme sea states

A deep learning super-resolution model to speed up computations of coastal sea states

J. Kuehn[a,*], S. Abadie[b], B. Liquet[c,d] and V. Roeber[a,e]

[a]Université de Pau et des Pays de l'Adour, E2S-UPPA, chair HPC-Waves, SIAME, Anglet, 64600, France
[b]Université de Pau et des Pays de l'Adour, E2S-UPPA, SIAME, Anglet, 64600, France
[c]School of Mathematical and Physical Sciences, Macquarie University, Sydney, 2109, Australia
[d]Université de Pau et des Pays de l'Adour, E2S-UPPA, CNRS, LMAP, Pau, 64000, France
[e]Department of Oceanography, University of Hawaii at Manoa, Honolulu, 96822, USA

ARTICLE INFO

Keywords:
Super-Resolution
Deep Learning
Coastal Wave Model

ABSTRACT

In this paper, the potential of a super-resolution technique is presented in the context of coastal wave forecasting. The method uses a neural network to predict a high-resolution spatial estimation of spectral wave parameters from a lower resolution numerical computation. In this particular example, one year of training data is sufficient to achieve satisfying accuracy for practical applications. The error of this method in reproducing the results of a high-resolution spectral model is an order of magnitude lower than the usual accuracy of spectral models. Simultaneously, it reduces the computation time by a factor of up to 50. Moreover, utilizing complementary training data of extreme events allows for a further improvement in accuracy. The study also shows that super-resolution is more accurate, albeit slower, than surrogate models, thus providing a trade-off solution between accuracy and speed. Overall, incorporation of the present approach into wave forecasting systems has the potential to rapidly generate "zoomed-in" areas of interest or to speed up ensemble forecasts without supplementary calculations at higher resolution.

1. Introduction

Many coastal communities rely on daily wave height forecasts for the purpose of safety and hazard mitigation. Over the last decades, improvements in numerical methods have lead to more accurate predictions of sea states, which had a considerable influence on maritime transport, fisheries, and ocean engineering (Gopinath and Dwarakish, 2015). Global efforts of producing ocean observation networks paired with national ocean services (e.g., the National Oceanic and Atmospheric Administration), national and international buoy networks (e.g., the CANDHIS network in France - `https://candhis.cerema.fr`), and global ocean wave models like SWAN - Simulating WAves Nearshore - (Booij et al., 1999) and WAVEWATCH III (Tolman, 2009) provide critical information - often in real-time. However, high-resolution (HR) data, particularly in the coastal zone, is often missing due to computational constraints associated with large computational domains in combination with fine meshes. Nevertheless, the need for local HR data still exists at this scale, since local forecasts for wave-driven processes are increasingly based on refined computations (Camus et al., 2011).

The recent rise in interest in Machine Learning and its ongoing integration into natural sciences can be largely attributed to its ability to rapidly perform various computational tasks after completion of an initial training phase. In

*Corresponding author
`jannik.kuehn@outlook.de` (J. Kuehn)
ORCID(s): `0000-0001-7678-54` (J. Kuehn)

this article, we present a super-resolution approach to decrease computation time for spectral wave forecasts by up to 50 times compared to traditional direct modeling of an equivalent domain at fine resolution. The concept relies on training a neural network to convert low-resolution (LR) coastal wave modeling results to a higher resolution for a specific study area. The main advantage of this approach is that the computation of a LR forecast and the subsequent conversion are considerably faster than an equivalent direct HR computation. This speed-up even grows for the increasingly common ensemble forecasting, where forecast results are based on multiple similar computational scenarios (O'Donncha et al., 2018). Furthermore, once trained, this model has the potential to be directly linked to the output of coastal wave models, thus providing local HR results on the fly without the need for costly direct calculations.

Super-resolution has been an active field of research in computer vision for almost a decade. Recently, it also started to be used in fluid mechanics - especially in the field of turbulence (Kim et al., 2019; Gao et al., 2021). In ocean sciences, deep-learning-based super-resolution was already successfully applied to sea-surface temperature of remote sensing data (Ducournau and Fablet, 2016; Su et al., 2021; Lloyd et al., 2022) or of modeled data (Thiria et al., 2023). Furthermore, treating gridded bathymetry data as digital images, Sonogashira et al. (2020) enhanced the resolution of coarse bathymetric charts and outperformed naive interpolation. This showcased that super-resolution might considerably reduce the amount of measurements needed. The idea of super-resolution in our study is similar to downscaling, where the spatial resolution of sea state variables is increased. Usually, this is done with either a dynamical approach using numerical wave models (Erikson et al., 2015) or a statistical approach based on empirical relationships between ocean variables (Hegermiller et al., 2017). Previous papers addressed the possibility of improving local forecasts with neural networks, for example, by correcting the wave model outputs with a neural network or random forests to better fit buoy observations (Bajo and Umgiesser, 2010; Londhe et al., 2016; Callens et al., 2020). Conversely, Lucero et al. (2023) successfully reduced WAVEWATCH III hindcast errors at critical locations with multi-layer perceptron and convolutional architectures, using a combination of the WAVEWATCH III model outputs and additional data. Apart from only improving forecasts or hindcasts, some authors trained neural networks to substitute the entire wave model, using directly the spectrum (James et al., 2018), bathymetry (Jörges et al., 2023), wind (Bai et al., 2022), and other data to forecast sea state variables like significant wave height. These surrogate models are usually much faster than the wave models, albeit less accurate, since all physical calculations are substituted by relatively simple parameterized matrix multiplications. Our super-resolution approach tries to complement this existing toolbox by introducing a tool that is considerably faster than the direct computations of a wave model and, simultaneously, more accurate than a surrogate model. To the best of our knowledge, this is the first study where super-resolution techniques based on neural networks are applied to coastal wave modeling.

In the present paper, we apply the Downsampled Skip Connections Multi-Scale (DSC/MS) neural network introduced by Fukami et al. (2019) to convert LR SWAN computations to a 16-times higher resolution. Section 2 describes the proposed workflow with a focus on data processing. Then, we present the results for an application of the model to a study case for the nearshore area at Biarritz (SW France), compare it to a surrogate model and analyze the possibility of decreasing the error for extreme events. Lastly, we discuss our results, the presently existing limitations of this approach, and the potential for future improvements.

2. Materials and Methods

The first step to construct a framework for neural-network-based super-resolution is to obtain a training and test data set from a coastal wave model.

2.1. SWAN and Data Pre-processing

To create a data set with matching LR and HR results, we compute various quantities including significant wave height H_s, mean wave period T_{m02}, and mean wave direction Dir with the third-generation spectral wave model SWAN for a part of the coastal area near Biarritz (see Fig. 1 left panel).

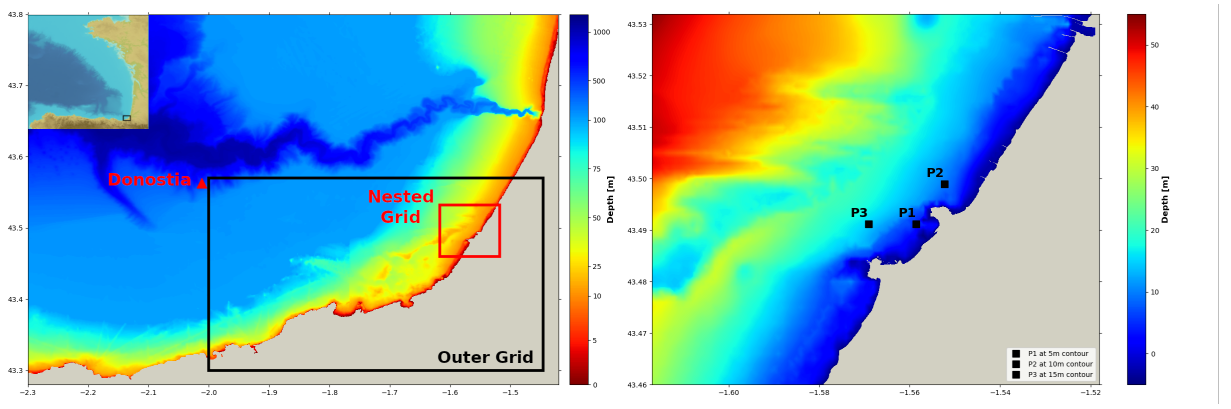

Figure 1: Left panel: Bathymetry map of the study area and the locations of the outer and the nested grid with a top left panel showing the location of the study site on the French Atlantic coast. Note that the scale of the colorbar is not linear to highlight bathymetric features over different scales. Right panel: Zoom-in on the nested grid. The three indicated points show the locations where the time series are extracted for the performance analysis in section 3.

The area of interest is nested inside a coarser outer grid that is forced by homogeneous spectral boundary conditions taken from the HOMERE hindcast database (Boudière et al., 2013), at the location of the Donostia buoy (i.e., around 35km from Biarritz in a water depth of about 450 m). On the right panel, the bathymetry of the nested grid is shown alongside with three locations, where we extracted time series of the sea state variables (see section 3). The locations were chosen to account for a bathymetry of 5 m, 10 m, and 15 m, for P1, P2, and P3, respectively. Note that for the panel on the left, the colorbar is not linear, for a better visualization of the bathymetric features over different scales.

The wave data set considered for this work covers a two-year time period spanning from January 01, 2018 to December 31, 2019. For the HR nested grid, we chose a grid with 160 x 160 quadratic cells (8 km in x- and y-direction with $\Delta x = \Delta y = 50$ m), which enables uncomplicated downscaling by multiples of two. To verify that our chosen resolution is fine enough to accurately capture wave processes, we performed a quick convergence analysis. For this we computed a stationary run for seven grid size steps, ranging from 12.5 m to 800 m. We then compared the mean values over the whole spatial domain for significant wave height, mean wave period, and mean wave direction (see Fig. 13 in the appendix). The results show that the difference between the mean values of a grid size step of 25 m and 50 m was under 0.5 % for each variable. Given the faster computation times and the lower computational load, we deemed 50 m sufficient for the HR nested grid to capture most variations in the wave regime. Furthermore, given that the bathymetry data has only a precision of $0.001\ ^\circ \approx 100$ m, a smaller grid size would not lead to a better representation of the seafloor. The LR grid that we convert to a higher resolution has 10 x 10 quadratic cells, with a $\Delta x = \Delta y = 800$ m. For HR and LR computations, we used a directional binning of 2.5° for a range of 215° to 55°. The frequency spectrum spans 0.05 Hz to 0.25 Hz and is divided into 25 logarithmically-spaced partitions.

The bathymetry data were obtained from the publicly available Digital Terrain Model "MNT bathymétrique de façade Atlantique" provided by the French Service Hydrographique et Océanographique de la Marine (SHOM) (SHOM-Service Hydrographique Et Océanographique De La Marine, 2015) and interpolated with the open-source program Octave (Eaton et al., 2020) to the two grid sizes.

With the objective to demonstrate the feasibility of the application of neural-network-based super-resolution to ocean wave modeling, we use basic, yet realistic settings for the SWAN computations. In particular, wind forcing, wind growth, and white-capping are not taken into account due to the strong influence of swells in the region. The water level was set constant and equal to the mean water level 2.25 m. Depth-induced wave breaking was modeled with constant values $\alpha = 1$ and $\gamma = 0.73$ adapted from Battjes and Stive (1985). Bottom friction is based on Madsen et al. (1988) with a constant coefficient of 0.085. A link to all SWAN run files, including the one for the outer grid can be found in the Data Availability section.

We note that the SWAN model returns NaNs (Not a Number) for cells that cross the shoreline. Since neural networks cannot work with NaNs, they have to be addressed separately as discussed in the workflow in subsection 2.4.

2.2. Neural network architecture

For our neural network architecture, we use an adapted version of the Downsampled Skip Connections Multi-Scale model, originally introduced by Fukami et al. (2019). We initially tested this model due to its success in reconstructing turbulent flows and its capability in capturing flow patterns over a wide range of scales. In the nearshore area, wave

transformation usually also occurs over a range of spatial scales - especially where the bathymetry gradients are locally large. This is particularly the case at the present study site, the Basque Coast, which exhibits very localized 3D bathymetry features in the shoaling zone. For instance, between Saint-Jean-de-Luz and Hendaye, embayed beaches, headlands, and breakwaters can be found over the course of just a few kilometers (Pinault et al., 2022). A particularly distinct feature is the large submarine canyon in the area around Capbreton (gouf de Capbreton). All these features have different scales, which reflect on the wave field. We thus expect a scale-adapted model, as the one proposed by Fukami et al. (2019), to work better than one that is scale-agnostic. Finally, as turbulent flows generally pose complex challenges to numerical solutions and exhibit more variance in time and space than spectral wave model computations, our choice of this neural network is rather conservative.

The DSC/MS model is composed of two parts, a submodel employing skip connections and max-pooling operations, that first down- and later upsamples the data; and secondly a so-called multi-scale model presented in Du et al. (2018). The skip connections are used to prevent degradation of deep neural networks (He et al., 2016), while the downsampling works effectively as a data compression, making the neural network more robust, particularly in the case of rotations and translations (Ngiam et al., 2010). On the other hand, the multi-scale model is composed of multiple convolutional filters of varying sizes to span different scales. The results of each submodel are finally concatenated where a final convolutional layer combines them into the reconstructed image.

For more detailed information about the model, we refer to the original article of the authors (Fukami et al., 2019) and more generally some introductions to the topic of convolutional neural networks (Guo et al., 2016; Rawat and Wang, 2017; Aloysius and Geetha, 2017). A flowchart of the DSC/MS neural network can be found in the appendix in Fig. 14 and the source code to train this model, as well as the following surrogate model, can be found in the GitHub repository. For more information, see the link in the Data Availability Section.

2.3. Surrogate Models

One alternative to the super-resolution approach is, as mentioned in the introduction, to use a neural network as a surrogate model. In a surrogate model, the entire spectral wave computation - in our case provided by SWAN - is substituted by a neural network. It usually takes the same inputs and is trained to reproduce the high-resolution results. The idea is based on the concept that the physical equations of the spectral wave model can be well approximated by a large parameterized model. A few recent examples of successful implementations of surrogate models are James et al. (2018); Huang et al. (2022); Mahdavi-Meymand and Sulisz (2023).

The advantage of a surrogate model over a super-resolution approach is that skipping the computation of a low-resolution result makes it even faster. However, the accuracy is strictly bound to the quality of the input data. Surrogate models have to work directly with wave spectra at the domain boundary, bathymetry, wind field, and other input data,

without having a first approximation of the actual wave height, period, or direction in the computation domain. Our super-resolution approach, on the other hand, receives exactly these features as an input or initial guess from a low-resolution computation of the SWAN model that, in turn, reduces the task of predicting the entire wave field from raw inputs to only adjusting it.

There is no single way of implementing surrogate models. James et al. (2018), for example, used a dense neural network, whereas Bai et al. (2022) and Huang et al. (2022) base their models on convolutional layers. To guarantee the same input for the neural network as for the SWAN computation, we choose an approach where the spectrum is flattened and transformed with a dense neural network to the same dimension as the bathymetry. The spectrum is identical to the one that is used as a boundary condition in SWAN consisting of 32 frequency and 24 directional bins. With the flattening operation the spectrum is reshaped to a vector with $32 \cdot 24 = 768$ elements that serves as input to the dense neural network. This dense neural network essentially upsamples the spectrum to 256,000 elements that are then reshaped into a 160x160 matrix - the same size as the bathymetry and the final output. Then, the transformed spectrum and the bathymetry are concatenated and passed to the same DSC/MS model as presented in the previous subsection 2.2. The flowchart for this neural network architecture can be found in the appendix in Fig. 15.

It is also possible to flatten both the spectrum and the bathymetry, to concatenate them and to pass this $32 \cdot 24 + 160 \cdot 160 = 256,768$-element vector directly to a dense neural network and only work with dense layers. However, the amount of parameters in this implementation grows very quickly ($>$ 100 million weights to adjust, in contrast to around 25 million) and we obtained inferior performances with this approach.

Note that for the following workflow and application, both the super-resolution and surrogate model were pre-processed and applied in the same fashion, except that we min-max-normalized the input for the surrogate model, whereas the data are not normalized for the super-resolution approach.

2.4. Workflow

First, we obtain the data set by computing a coastal wave model over two grids - a fine and a coarse one - corresponding to the high-resolution ground truth and the low-resolution input. As we will discuss in subsection 3.1, it is possible to specifically generate extra data to mitigate the prediction errors in certain wave regimes. Then, we split the data into "snapshots", i.e. into files containing the spatial distribution of a variable (e.g., a 2D H_s map) at one time step. In our case, the sampling interval is one hour in both the high- and low-resolution data set. As a next step, a certain fraction of the whole data set is set aside for testing purposes. The rest of the data is divided into a training and validation set. The former is used by the model to adjust the weights, whereas the latter is for performance assessment of the model after each training epoch. The NaNs resulting from the topography part of the coastal wave

model computations are set to zero, as the neural network handles only numbers. We train the models with ADAM optimization (Kingma and Ba, 2015) and early stopping after 30 epochs (Prechelt, 1998). Finally, we evaluate the neural networks on the test set to obtain a realistic estimation of their performance. Fig. 2 summarizes our approach.

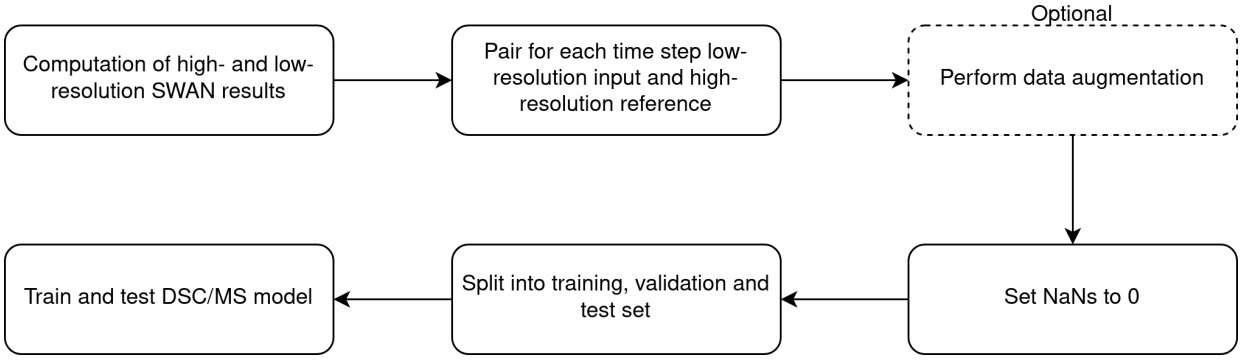

Figure 2: Workflow of the super-resolution approach adopted in the present study.

Note that, in contrast to Fukami et al. (2019), the networks are trained by using the L1 / mean absolute error (MAE) loss instead of the L2 / mean square error (MSE) loss. This is due to the fact that the L2 loss penalizes larger and tolerates smaller errors. In super-resolution tasks, this commonly leads to overly smooth results (Wang et al., 2020).

Also, Fukami et al. (2019) used average- or max-pooling to artificially create the LR input. However, a model trained on this input will not necessarily perform well on real LR results, since in practice LR computations are rarely simply an average-pooled version of HR computations. Thus, we decided to train the model with real LR - HR pairs, stemming from practically the same SWAN computation, but at different resolutions.

2.5. Application

In the present study, we run a pair of high- and low-resolution SWAN computations for the two-year period between January 01, 2018 and December 31, 2019. We use the first year for training (80 %) and validation (20 %), and the second year solely for testing. Using a whole year for testing helps to get a realistic estimation of the performance of the model over various sea states. Finally, for each of the three variables - significant wave height, mean wave period, and mean wave direction - we train a model to convert low-resolution SWAN modeling results from a grid cell count of 10x10 to one of 160x160.

3. Results

As an initial sanity test, we compare the DSC/MS model to two less adapted and simpler architectures: a Fully-Connected Neural Network (FCNN) and a Convolutional Neural Network (CNN).

Architecture	# Parameters	Min. MAE [cm]	Max. MAE [cm]	Mean MAE [cm]
LR Input	-	-	-	11.29
CNN	19k	7.85	8.23	8.00
FCNN	286,000k	0.79	1.23	0.98
DSC/MS	134k	**0.61**	**0.85**	**0.71**

Table 1
Minimum, maximum, and mean MAE over 6 training runs of different neural network architectures, computed over the test data set. For comparison, the MAE of the raw low-resolution input in comparison to the high-resolution reference is displayed too.

We use a FCNN of two hidden layers with a Rectified Linear Unit (ReLU) activation. While deeper models are more common, they are computationally hard to implement in the given case, since the amount of parameters increases rapidly due to the desired high-resolution output. Here, with only two hidden layers, the model contains over 286 million parameters, requiring 3 GB of memory.

On the other hand, the CNN is composed of an initial upsampling layer, 3 hidden convolutional layers with a kernel of 3, a filter size of 32, padding to keep the output at the same size, and ReLU as an activation function. The last layer has only one filter (for the high-resolution prediction), a kernel of 3, padding, but no activation function. This simple architecture drastically reduces the number of parameters to only around 19 thousand. The architecture of this CNN was chosen to be similar to a classical baseline model for a simple machine learning task, like classifying the handwritten digits of the MNIST data set (Deng, 2012).

The proposed DSC/MS model situates itself in the middle, with 134 thousand parameters and a more elaborated architecture. As a comparison, the model is of much more manageable size with around 2 MB.

For each architecture, we train the model six times and compute the mean absolute error over the test data set. The minimum, maximum, and mean performance for each architecture are presented in Table 1. A more detailed explanation of the implementation along with flowcharts for the FCNN and CNN can be found in the appendix in section A.1.

While the CNN does have the lowest amount of parameters, it is not efficient enough for a super-resolution task, the results improving only slightly the low-resolution input. The FCNN and DSC/MS are much closer in performance, however, the latter does not only fare better, but does so with a fraction of the amount of parameters.

Nevertheless, we note that the DSC/MS architecture is certainly not the only architecture that is capable of performing super-resolution. Generative Adversarial Networks (GAN) (Stengel et al., 2020) are one example of a potentially more powerful architecture, but they come with an increased difficulty of implementation and longer training times. We choose here the DSC/MS model for its balance between complexity, speed, and its problem-specific architecture mentioned in 2.2.

Before analyzing the performance of the super-resolution and surrogate neural networks in more detail and over different sea states, we compare them in terms of MAE computed over the whole test data year and over the whole domain. Note that due to this double-averaging operation, the obtained value is a global indicator, which hides the temporal and spatial variations of the error. For each variable, we trained each architecture multiple times to get a rough idea of the range of performances and to consequently be able to make a more robust comparison.

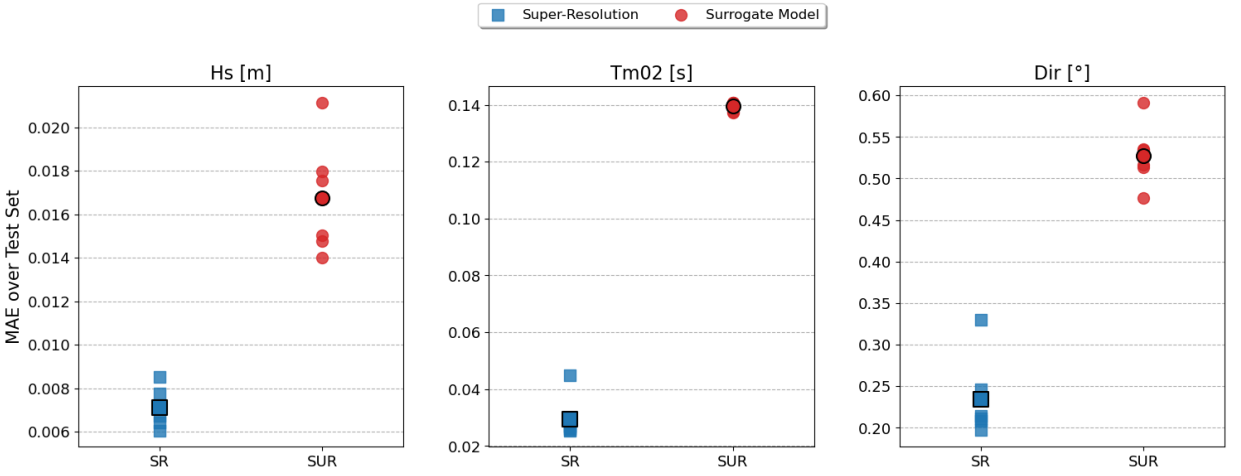

Figure 3: Mean absolute errors for the super-resolution and surrogate models computed over the test year for the three sea state variables. Each model was trained 6 times, the outlined data point is the mean of the errors.

Fig. 3 presents our results. While both approaches are on average able to predict accurately the wave parameters, there is a clear discrepancy between the two. The best surrogate model performs worse than the worst super-resolution model for each of the three variables. For example, the best H_s surrogate model has a MAE of 1.40 cm, in comparison to 0.85 cm for the worst DSC/MS model. The mean errors, highlighted in black, are around 2-times lower for significant wave height and mean wave direction, whereas more than 4-times lower for mean wave period. While the absolute value changes seem relatively small - for instance, an improvement from 2 cm to 1 cm for H_s - they only reflect a mean change, as mentioned above. Such changes usually encompass larger error reductions along the coastline, as we will see below.

For all of the following results, we selected, for each variable, the worst training run to avoid overly optimistic predictions. Also, all of the figures can be found, interactively, in Jupyter notebooks (see the Data Availability section for more information).

In Fig. 4, we give an example of the DSC/MS predictions of the spatial distribution of significant wave height, mean wave period, and mean wave direction. Given the similarities of the results of the surrogate model in this wave regime, we only focus on the super-resolution model results. The corresponding plot for the surrogate model can be

found in the appendix (Fig. 16). For comparison, the LR input and the HR ground truth are displayed as well. Here the chosen data instance is representative of the average wave regime, which corresponds to H_s values of 1.0 m to 1.5 m, T_{m02} around 7 s to 10 s, and an incoming mean wave direction in the range of 300° to 320°. Furthermore, the last row of the figure shows the difference of the high-resolution computation and the neural network prediction for the whole domain.

In the three cases, the high-resolution computations are well approximated by the neural networks. The prediction not only correctly captures most of the wave features, but also reconstructs the original coastline in a sense that the neural network predicts negligible values for values on land - setting values lower than a small threshold (around 0.01 for all variables) to NaN results in a near-perfect reconstruction of the original coastline for each of the models.

Out of the three variables, the significant wave height and the mean wave direction (MAE of 0.66 cm and 0.23°, respectively) are reproduced best. The 2D error map for H_s shows that there are some minor discrepancies on the scale of 1 cm over the whole domain, with some larger errors of around 5 cm along the coastline, usually concurrent with wave refraction patterns. On the other hand, the error for the mean wave direction is much more localized, being on the order of 10° for only some locations where wave direction changes quickly, but much smaller (approximately 0° to 1°) for the rest of the domain. When comparing the performance over the whole domain, the mean wave period has a lower absolute value of the mean average error (0.07 s) than the mean wave direction. However, these values cannot be compared over different variables, due to their different scales and units. The normalized root-mean-square error (see Equation (1) below) allows a more robust comparison, since it is based on percentage points and is normalized by the mean of the data instance. Here, the mean wave direction is predicted with the highest accuracy, followed by H_s, and T_{m02} (0.58 %, 0.92 %, and 1.21 %, respectively).

For the mean wave period, minor reconstruction errors are visible, particularly in the upper right-hand corner around the location of an artifact of the boundary condition and at the wave refraction patterns, which are reproduced, to a certain degree, by the 2D error map. At these locations, the difference reaches absolute values of 1 s, but the error throughout the domain is usually much lower, in the range of 0.1 s to 0.5 s. Furthermore, in contrast to H_s and Dir, the reconstruction is not as sharp as the ground truth. Note that this is a data instance were the neural network prediction is very accurate over all three variables. It is the case for the majority of the test data, but in certain conditions the performance of the neural network drops significantly. In the following figures we analyze in more detail these conditions.

First, to determine the accuracy of the method over time, we extract three time series for each sea state variable at the three locations mentioned in subsection 2.1 over the whole year of test data. Fig. 5 shows an extract of the

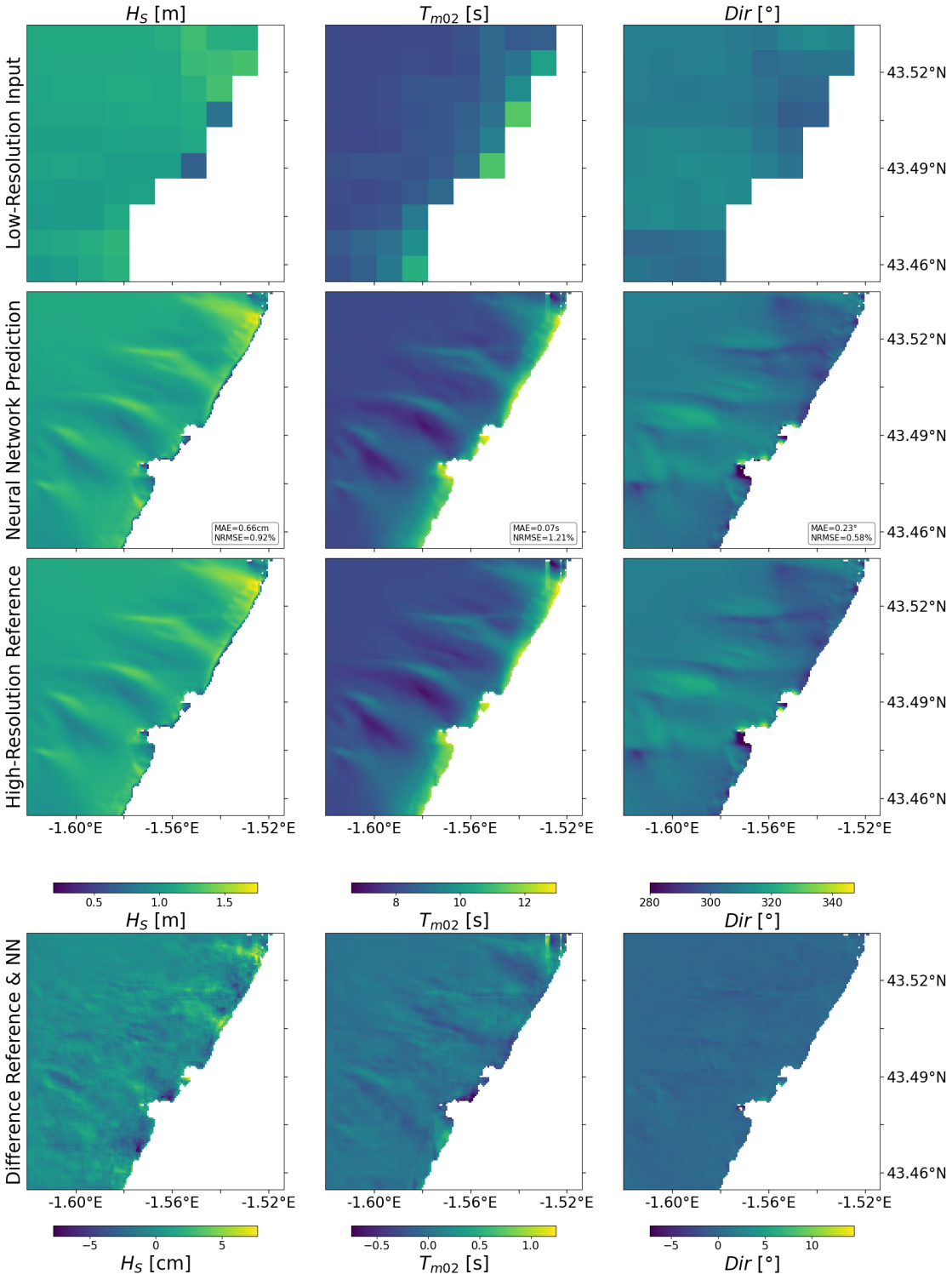

Figure 4: Neural-network reconstructions of high-resolution computations from lower resolution SWAN calculations. The MAE and NRMSE (see equation (1)) is shown for the predictions. Additionally, the fourth row shows 2D error maps of the high-resolution SWAN computation and the neural network prediction. The data instance is from April 21, 2019 at midnight.

time series during an extreme wave event at the location P2. The whole time series and the other locations can be found and visualized in the Jupyter notebooks. In general, both models perform well, improve upon the low-resolution computation, and usually give predictions close to the reference. Nevertheless, the super-resolution approach exhibits a smaller discrepancy, sometimes reaching better performances on the scale of tens of centimeters, in the example of significant wave height. In this extract particularly, the surrogate model has difficulties correctly predicting the wave height at the storm peak, but also, more globally, the mean wave period and direction. An example of the advantage of the super-resolution approach can be seen at the prediction of the mean wave period around the 20^{th} of December. Here, the low-resolution input provides already a very good estimate of the high-resolution ground truth and consequently the super-resolution neural network only has to adjust the input slightly. The surrogate model on the other hand has to "deduce" all the physics by itself, which in this case, leads to a distinctly poorer prediction of almost 1.5 s of difference.

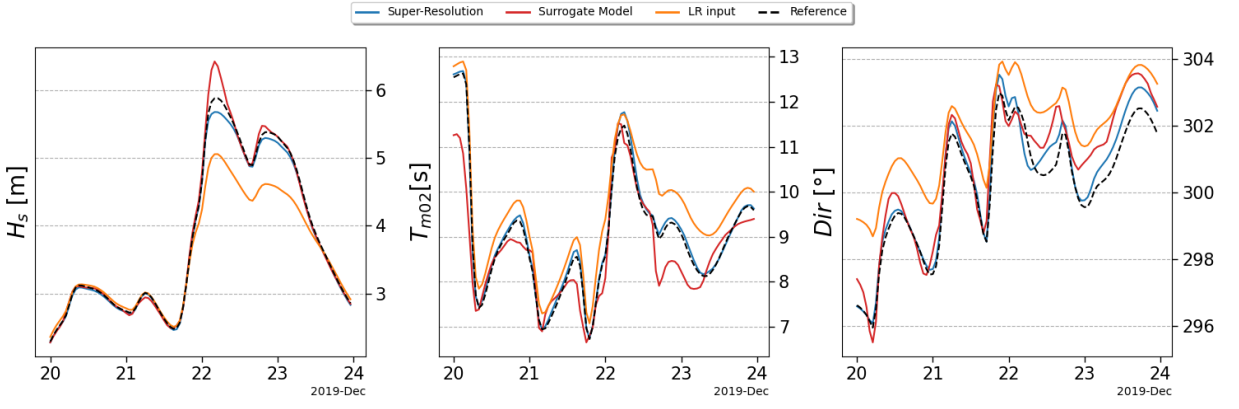

Figure 5: A 4-day time series extracted at location P2 (located at a depth of 10 m). The time series covers a storm event around the 22^{nd} of December. The whole time series and the other locations can be found in the Juypyter notebooks.

For a more thorough comparison over the whole time series, we compare the predictions to the HR ground truth as well as the LR input. We also compute various commonly used quantities of interest to aid the interpretation. These include the root-mean-square error RMSE, the normalized root-mean-square error, also sometimes called scatter index NRMSE, and the bias. They are computed as follows

$$\begin{aligned} \text{RMSE} &= \sqrt{\sum_i \frac{(y_i - \hat{y}_i)^2}{n}} \quad \text{NRMSE} = \frac{\text{RMSE}}{\bar{y}} \\ \text{bias} &= \sum_i \frac{y_i - \hat{y}_i}{n}, \end{aligned} \tag{1}$$

where y_i the i-th point of the ground truth, $\bar{y}$ the mean ground truth over the whole time series, $\hat{y}_i$ the corresponding prediction by the neural network, and n the total number of data points. The scatter plot of the time series and their corresponding quantities are displayed in Fig. 6 for both the super-resolution and surrogate models.

Starting with the left-hand column, the significant wave height is reconstructed quite accurately with a MAE of 4 cm or less for both the DSC/MS and surrogate model, and deviates from the reference mostly only at both extremes. A deterioration at very low values is seen at P1 as a downward-curved tail, where the model underestimates the wave height. At the locations P1 and P2, the low-resolution computation diverges quite strongly from the high-resolution reference for larger wave heights, being up to 2 m off in extreme cases, which consequently affects the super-resolution predictions as well. At the location P1, the significant wave height is predicted worse than with the surrogate model, mostly due to this large error. Nevertheless, both RMSE and NRMSE are still close to the values of the surrogate model. Additionally, for the other two locations the error is distinctively smaller and the super-resolution approach outperforms decidedly the surrogate model, approximately cutting the error in half - namely a RMSE of 1.51 cm compared to 3.97 cm at P2, and 2.62 cm instead of 3.98 cm at P3 for the super-resolution and surrogate approach, respectively.

In the middle column, the results are presented for the mean wave period, which is predicted reasonably well by both models, the NRMSE generally being under 6 %. Indeed, particularly the super-resolution model shows a significant improvement over the low-resolution input and exhibits for all three locations a NRMSE of only around 1 %. In contrast to wave height, the largest spread is found mostly around the middle, with data points at both extremes being relatively close to the ground truth. This stands in contrast with the amount of training data, that is sparser at the extremes. We presume that the larger spread stems from an overall larger average value (compared to wave heights), and a larger amount of wave periods in the same range that, consequently, also contain statistically more data instances that are harder to predict. Interestingly, the surrogate model is markedly less accurate for the mean wave period, where in some cases the error is higher than for the LR computation. This might be due to the fact that significant wave height is greatly influenced by bathymetry, which thus provides a valuable input. While mean wave period (and direction) do also depend on bathymetry, the influence is less strong and the input is less valuable, leading to a deterioration in the predictions.

Finally, in the case of the mean wave direction, the large absolute values of the deviations, of up to 10° to 15°, and the multiple loop-like structures, give the impression that the neural networks had the largest difficulties with this variable. It should be noted, however, that larger deviations are also partly due to larger overall values and the comparatively large range of values that it spans (more than 60°, compared to only 6 m in the case of significant wave height). In fact, the NRMSE is the smallest of the three variables, with a maximum NRMSE of less than 0.6 %, pointing to a very good performance of the neural networks for the reconstruction of the mean wave direction. Nonetheless, the time series do exhibit some clear discrepancies, usually induced by inaccurate inputs for the super-resolution (SR) approach, as in the case of the plot for the location P1, for example, where the input is almost constantly deviating

by 10° to 20°. All plots show the occurrence of loop-like structures at larger mean wave directions in the data, that correspond to specific events in the time series. Accordingly, those structures in the input are reflected in the model prediction and lead to a collection of data points that are reconstructed less accurately. These loop-like events occur when the whole spatial domain is dominated by incoming waves from uncommon directions. Given that these events are rare (and relatively short in time) in our domain, the neural networks do not have enough similar data to correctly reconstruct these cases.

To have a more quantitative view on the reconstruction errors and performances in different regimes, we divide each range of values of the variables into 12 bins and compute the normalized root-mean-square error (NRMSE) like in equations (1) and mean difference $\frac{1}{n}\sum_i^n |y_i - \hat{y}_i|$ for each bin. This is done for the three locations P1, P2, and P3. In Fig. 7, we display the NRMSE, the mean differences and their standard deviations and, for an easier interpretation, the distribution of the training samples that the neural networks trained on. To better illustrate the variations of the NRMSE over the three variables, the y-axes are not scaled. It is however important to bear in mind that the magnitudes of the error, i.e., the maximal NRMSE of significant wave height, is considerably greater than the size of the maximal NRMSE of the mean wave direction. The figure presented in the main text is based on the results of the super-resolution model. An equivalent plot for the surrogate model can be found in the appendix in Fig. 17.

For the significant wave height, we confirm exactly the same pattern as in Fig. 6. The overall error is quite low, especially for the locations P2 and P3, where the mean difference does not even exceed 20 cm, even for larger wave heights. Overall, the error grows generally inversely proportional to the available training data, namely increasing for larger wave heights and, to a certain extent, for very low wave heights. Noteworthy is that the NRMSE has its minimum for the three locations at around 3 m to 4 m, which does not coincide with the maximum amount of training data. This can be explained by the fact, that the mean difference stays constant for a large range of values. Thus, with larger or smaller values of y_i, the error decreases or increases as well, without the model necessarily being more or less accurate. More specifically, even though the NRMSE values are relatively high for lower wave heights, the actual difference between the prediction and the high-resolution reference stays still negligibly small over the whole range of lower wave heights.

The trends are not as clear for the mean wave period, where a large amount of training data does not always correlate with low differences or errors. Given the rather large standard deviations, however, it is difficult to reach a conclusion, especially due to the fact that the amplitudes of the variations of the differences are only on the order of 0.05 s, except for the location P3.

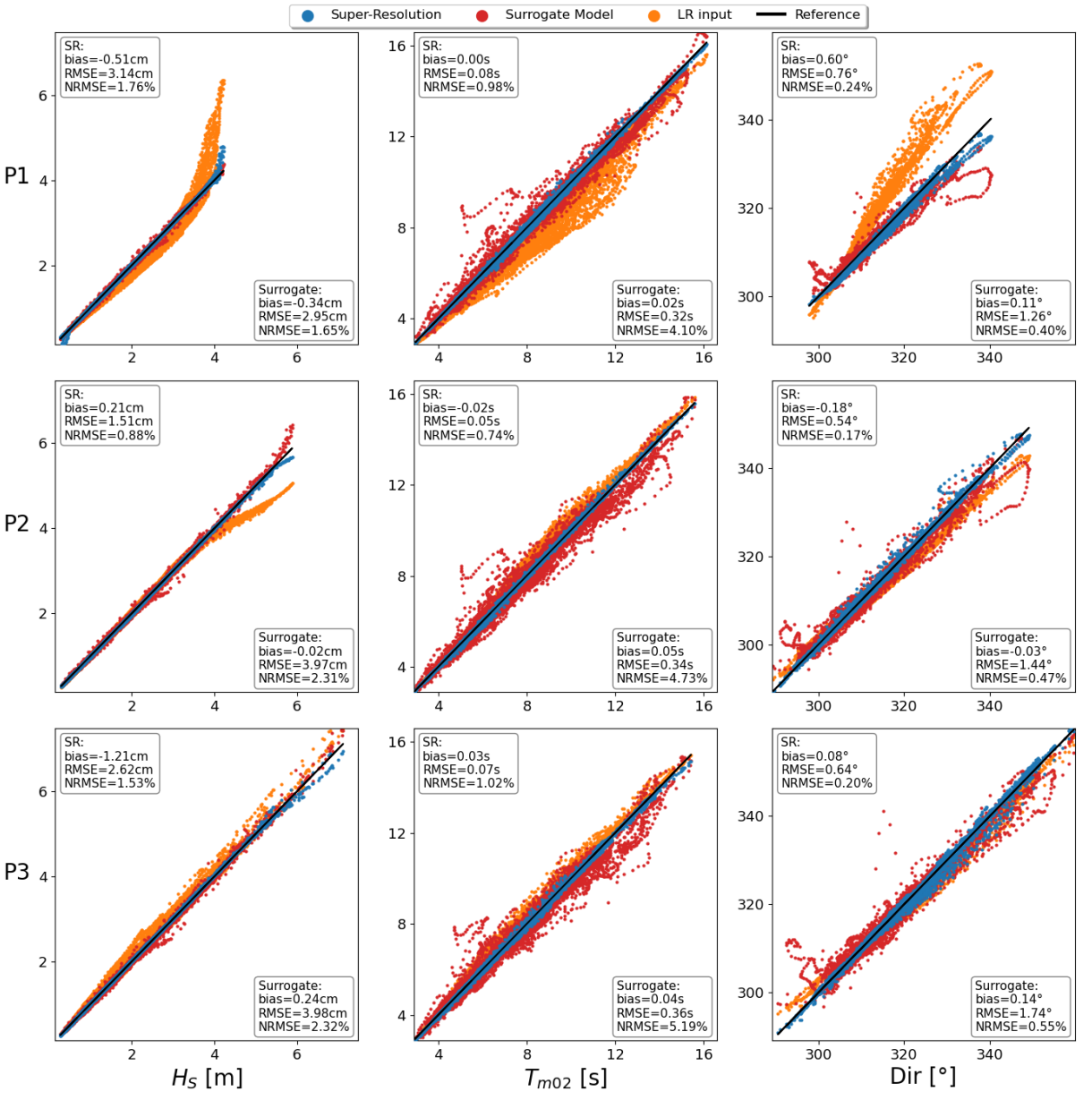

Figure 6: Scatter plots of low-resolution input and predictions of the two neural networks. Each subplot shows the result of the low-resolution computation (orange), prediction of the super-resolution model (blue), and prediction of the surrogate model (red) against the high-resolution reference for each variable (H_s - first column, T_{m02} - second column, Dir - third column), at each location P1 (5 m depth, top-row), P2 (10 m depth, middle), and P3 (15 m depth, bottom-row). The black diagonal corresponds to an absolute match with the high-resolution reference.

As for the mean wave direction, the trend of the difference seems to generally agree well with the training data distribution. The drop at the far end of the curves, where both the NRMSE and mean difference drop again after an initial rise is an unexpected pattern, given the low number of observations. This might again be due to large standard deviations or patterns that are easy to pick up for the neural network even with a small amount of training data.

Note that altogether the error is low for all three variables over a large range of values. In the case of H_s the NRMSE is less than 2 % for more than 95 % of the data - for T_{m02} and Dir this amounts even to 100 % of the test set. Indeed, in the case of the mean wave direction, the mean difference is smaller than the resolution of the SWAN computation in itself. We elaborate this point further in the discussion.

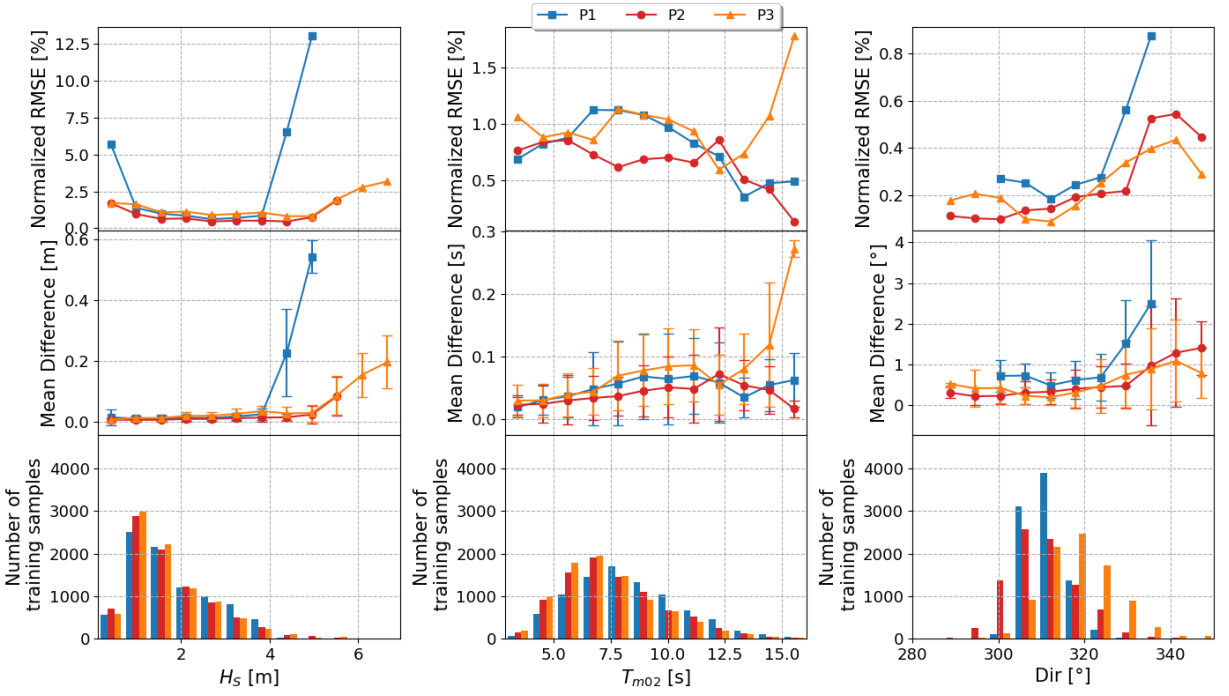

Figure 7: Normalized root-mean-square errors and mean differences with their standard deviations over 12 bins. The three colors indicate the different locations P1, P2, and P3.

3.1. Data augmentation

With the help of the two previous figures, we found that especially for significant wave height, a lack of data of the extreme cases seems to be linked to worse predictions of those sea states. We tried to improve the predictions of the super-resolution approach by artificially increasing the size of the data set by adding new data of extreme cases to the training set. This *Data Augmentation* approach is a common procedure, especially in the area of image recognition (Perez and Wang, 2017). Usually, to enhance the size of the data set and to improve the robustness of the neural network, images are transformed in various ways and then added to the original data set. As an example, a picture of an object can be mirrored horizontally or the brightness and contrast of the image can be changed, but the object in the picture would still be the same. These artificial modifications help the neural network generalize better and consequently boost performance (Shorten and Khoshgoftaar, 2019).

In this study, we increased the number of training data at extreme wave heights by running complementary stationary LR and HR SWAN computations. The following thresholds are strongly dependent on the study case and

should be used with caution and chosen accordingly to the given data set - we achieved good results with the following procedure as summarized in the flowchart 8:

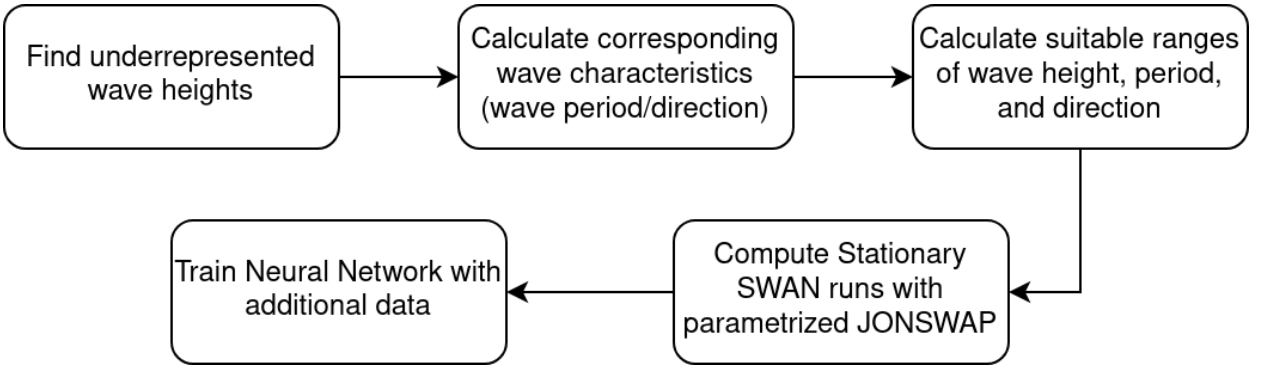

Figure 8: Flowchart for the proposed data augmentation workflow.

First, we compute the significant wave height that is larger than 95 % of all wave heights. We then compute the range of the corresponding wave period, and wave direction at those extreme wave heights. Using these three variables, we can parameterize a JONSWAP (Joint North Sea Wave Project) spectrum in SWAN to run a stationary computation. As we are here focusing on improving the prediction of significant wave height, we vary this parameter the most - 60 values ranging from 4.5 m to 9.0 m. Wave period ranges from 13 s to 16 s in 5 values, and lastly, wave direction from 300° to 330° in 3 values, resulting in a total of 900 extra data points, which is a little more than 10 % of the original data set size.

To detect the differences in performance for wave heights in various regimes, we compare the data augmentation approach to the previous scatter and error plots. In the case of the scatter plot, we focus especially on the larger wave heights, by zooming into the window with the 5 % largest wave heights. Fig. 9 shows that the effect of the data augmentation depends strongly on the location. At the first location (left-hand subplot), the data augmentation approach decreases by a factor of three to four the bias, RMSE, and NRMSE, as can be verified in table 2. A slight improvement is seen also at the two other locations, even though the effect is much subtler - a decrease of only a few percentage points in NRMSE - partly due to the already good results. Even when compared over all wave heights, data augmentation still improves overall performance, although less significantly (see appendix Table 3).

For a more nuanced picture of the error in different wave regimes, we recompute an error plot, again dividing the range into 12 bins. The result is shown in Fig. 10. We find similar trends as for the scatter plot (Fig. 9). At location P1, the improvement with data augmentation is clearly visible, the mean difference reduces from almost 60 cm to 5 cm. On the other hand, at P2 and P3 the decrease is less distinctive. Interestingly, at P1 the error decreases even for small wave heights, even though no additional data were used in this regime. Conversely, the error slightly increased for P3

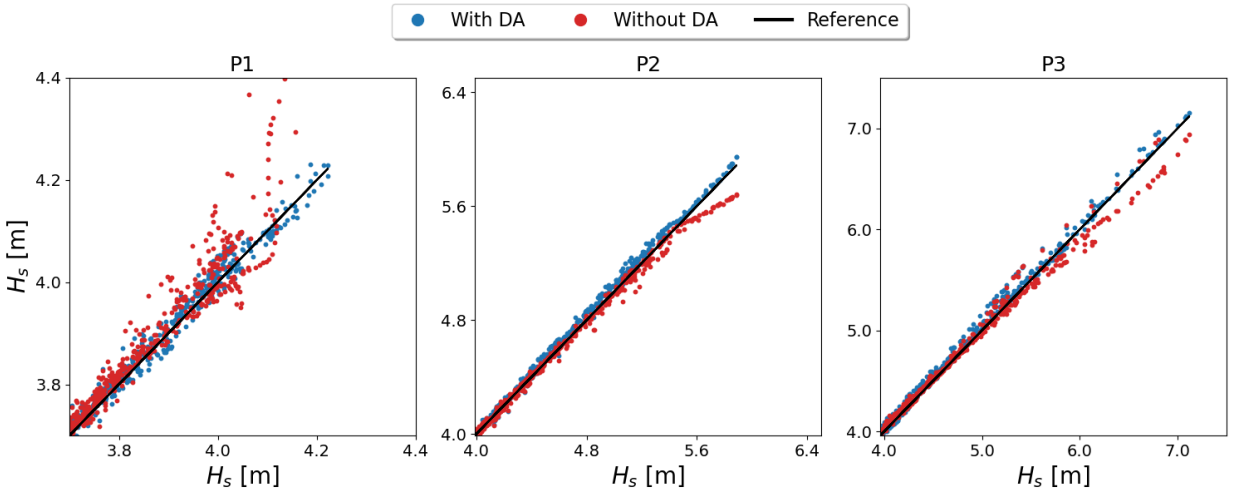

Figure 9: Scatter plot of the neural network H_s predictions against the high-resolution SWAN results for networks trained with and without Data Augmentation (DA) at different locations. Only the 5 % highest wave heights are shown. Note that the points without data augmentation (in red) correspond to the predictions of the super-resolution approach in the first column in Fig. 6.

Location		Bias [cm]	RMSE [cm]	NRMSE [%]
P1	With DA	**-0.9**	**2.1**	**0.5**
	Without DA	-4.1	6.5	2.9
P2	With DA	-3.2	**4.2**	**0.9**
	Without DA	**1.5**	4.6	1.0
P3	With DA	-4.1	**6.5**	**1.3**
	Without DA	**0.0**	8.0	1.6

Table 2
Bias, RMSE, and NRMSE for time series computed at locations P1, P2, and P3 for neural networks trained with and without data augmentation, for the 5 % largest wave heights.

at small wave heights, which can be explained by the fact that the neural network tries to generalize over a large range of wave heights and thus performs slightly worse in certain ranges.

3.2. Depth-dependent errors

Given that our study area is strongly influenced by bathymetric features, we analyze the variation of the absolute difference of H_s between the neural network predictions and the high-resolution reference dependent on water depth. To achieve this, we split the range of bathymetry values into 20 equal bins and compute the mean absolute difference in each of these bins, over the whole test year. This is done not only for both the SR and surrogate model, but also for the low-resolution input for comparison. The results are displayed in Fig. 11 along with binned mean H_s values of the high-resolution computation (error bars represent the minimal and maximal H_s value over the test year), as well as a water depth histogram to aid interpretation.

In low water depths, both architectures exhibit very small mean differences in the range of 0.5 cm to 1.5 cm. At a water depth of around 5 m, the curves start to diverge. While the super-resolution curve peaks at approximately 1.7 cm

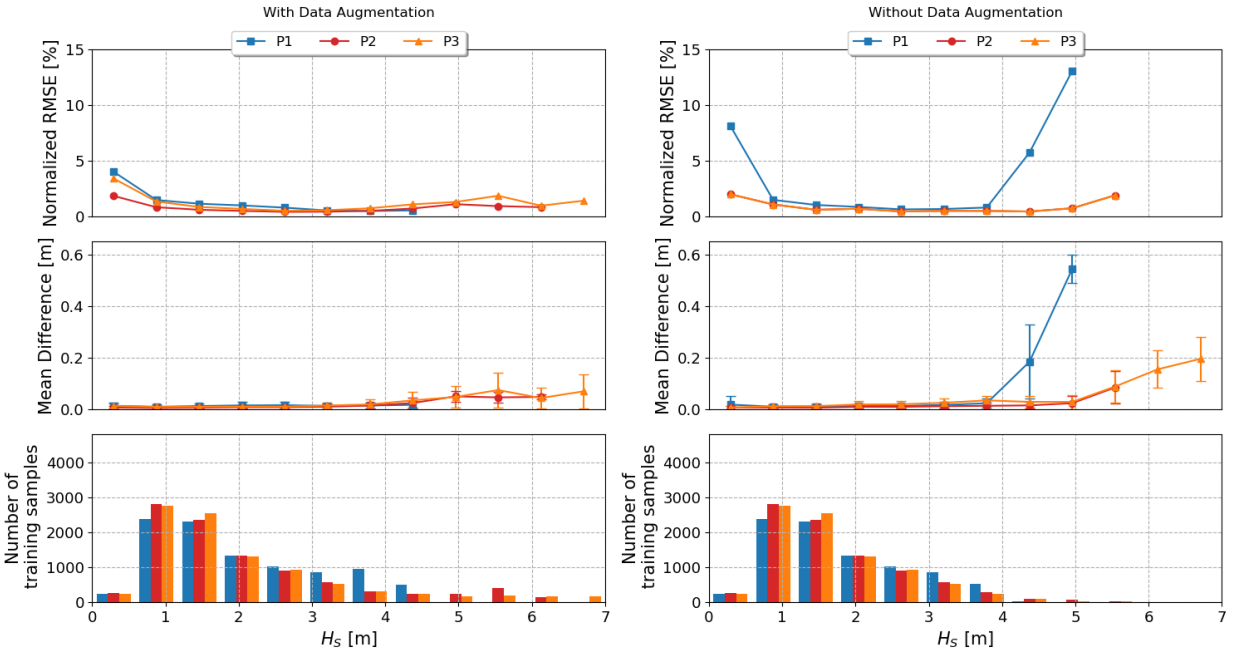

Figure 10: Normalized root-mean-square errors and mean differences of the significant wave height predictions, compare with Fig. 7. The left column shows the values with and the right column without the data augmentation approach. Note that the right column is identical to the first column in Fig. 7

after which it falls almost continuously, the surrogate model curve stays relatively constant at a difference of 2 cm, dropping only slightly at much larger water depths. While the order of magnitude is much larger, the low-resolution curve resembles somewhat the one of the super-resolution model. It starts at a peak of around 80 cm, before decreasing drastically to roughly 1 cm, with a small upward dip towards the end.

The very low differences along the coastline can be explained by two factors: First, the significant wave height has smaller overall mean values very close to land, as can be seen in the lower-left panel of Fig. 11, and consequently the differences are smaller as well. Second, the neural network learns the coastline through the repeated exposure, since it is one of the constant features throughout the data set, and is able to reconstruct it exceptionally well. However, the errors start increasing quickly, due to high biases through the low-resolution input, but also due to larger variances (note how the maximal H_s values increase until around 27 m) and more complex phenomena like shoaling and refraction that have to be taken into account by the neural networks. The large discrepancy at greater water depth between the two models is interpreted by taking the type of input into account. Although the surrogate model has the full spectrum and the bathymetry as an input, the neural network needs to do the physical conversion itself, which is inherently more difficult than the SR model's work of adjusting the - at lower water depths - already accurate low-resolution input. Both models exhibit an improved performance offshore, due to a reduced variance of the wave heights, but also owing to the much less complex wave field.

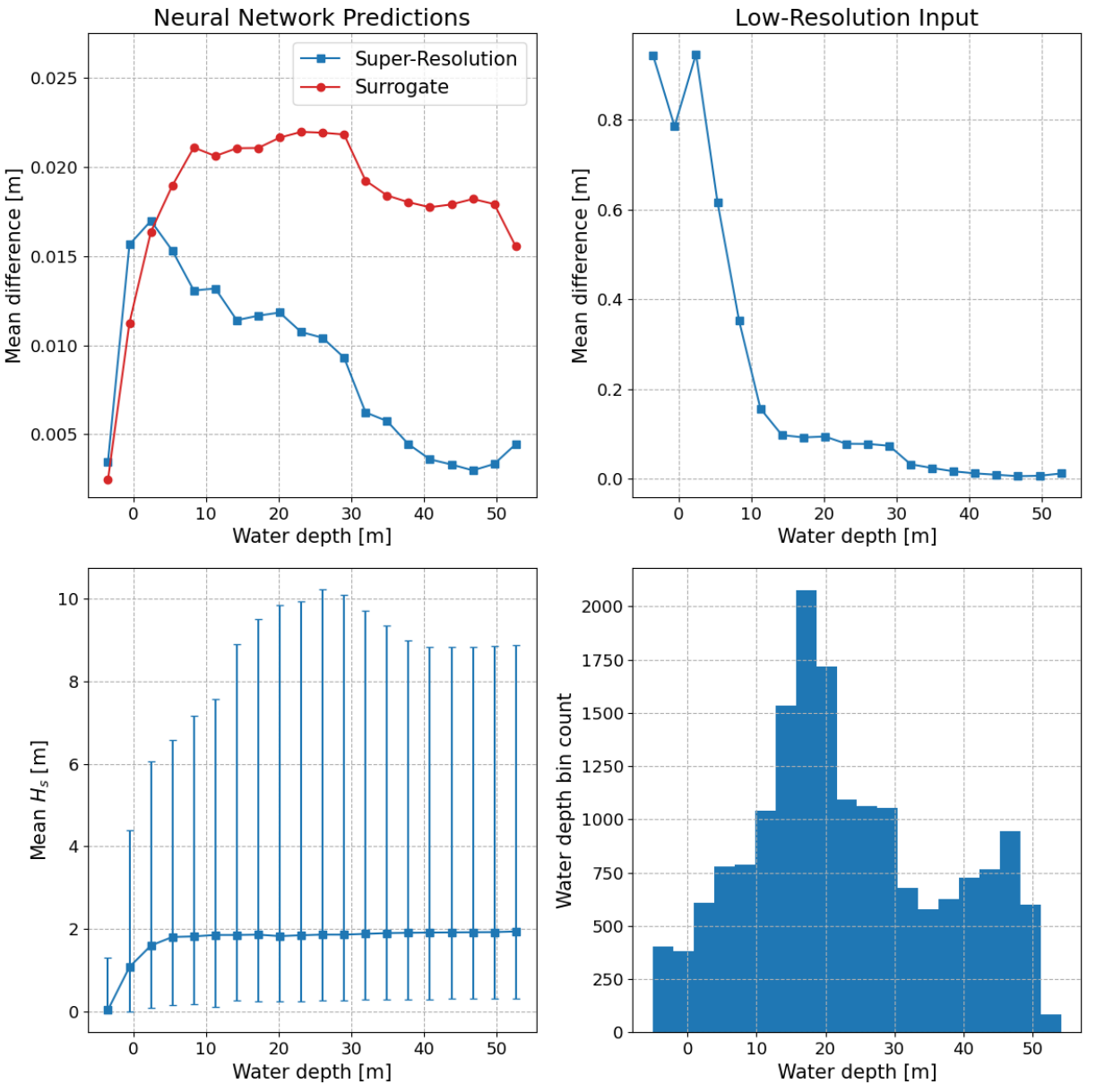

Figure 11: Mean absolute differences of neural network H_s predictions and high-resolution ground truth in dependency on water depth for both model architectures. For comparison, the difference is also shown for the low-resolution input. Additionally, as support for interpretation, the mean H_s values of the high-resolution computation for a certain depth are given, with minimal and maximal values as error bars, as well as a water depth bin count.

Lastly, while the frequency of how often a certain water depth exists could have an influence on the results, especially for the surrogate model that uses the bathymetry directly as an input, there seems to be no such apparent link. The shape of the histogram follows somewhat loosely certain variations of the mean differences of the surrogate model, but the similarities are too small to deduce a clear correlation between them. We still provide the water depth histogram as additional information about our study area.

3.3. Comparison of Computation Time

For a better evaluation of the computational performance, we compare each needed step in a surrogate model and our super-resolution approach to the computation time of a traditional direct HR SWAN calculation. The SWAN computations were run with 12 parallel threads on an Intel Core i7-9750H processor with 6 cores and 12 threads. For pre-processing, we used the same processor without parallel threading and for training and prediction of the neural network, a NVIDIA GeForce RTX 2070 Max-Q was used.

As a first step, the outer grid of the SWAN model was computed over the time range of 2 years, which took approximately 7.5 h. Similarly, computing the same time range in the nested high-resolution 160x160 grid took also 7.5 h. Calculation of the same nested grid in lower resolution (10x10) required only 8 min.

Training times of the neural network can change significantly for the same model if the algorithm gets stuck in a local minimum. For the models presented, the times required were around 2 h to 4 h, which adds up to approximately 9 h for all three variables - that is, more than the HR SWAN computation. Another factor that has to be included, is the time required to create the training (and potentially test) data set, which with the outer grid, HR, and LR computation included, sums up in this case to a bit more than $7.5\,\mathrm{h} + 7.5\,\mathrm{h} + 8\,\mathrm{min} \approx 15\,\mathrm{h}$. This is already more than twice the high-resolution computation time itself, without yet taking into account the training time. One has to note, however, that both setting up the data set and the actual training of the neural network are only a one-time cost, as the models can be reused multiple times after the one-time training. Additionally, if available, it is possible to use an existing high-resolution hindcast, so that only the complementary low-resolution data set has to be created, resulting in a distinctive reduction of the upfront computation costs. In the following, when addressing the matter of an improvement in computation time, we refer solely to the execution of the neural network, when it is already trained, i.e., without the upfront computational costs.

The actual prediction by the models is computed very fast, since once the weights are determined after training, predictions are mainly a form of matrix multiplications. Converting two years of data on the GPU required only around 30 s, both with the surrogate and super-resolution model. In our approach, together with the time of running the LR SWAN computations, the overall time amounts to 8.5 min. Compared to the 7.5 h for modeling the domain directly in high-resolution, the gain in computation time is more than 50-fold. Even if the prediction of the neural network is done on a CPU, in a case where the model is already trained, but no GPU is available for prediction, the increase in computation time is considerable. On the CPU converting 2 years of data takes around 9 min, which with the LR SWAN computation of roughly 8 min results in around 17 min - still a more than 25-fold speed-up.

The speed-up of the surrogate model is even greater, since no low-resolution computation is needed. Given that the computation only takes 30 s on a GPU for the two years, the model is 900-times faster. On the CPU the prediction takes around 10 min, which still is equivalent to a 45-fold speed-up.

4. Discussion

Our adaptation of the DSC/MS model proposed by Fukami et al. (2019) is able to reconstruct high-resolution features of various sea state variables with only a low-resolution SWAN calculation as input. It is a considerable alternative to surrogate models, acting as a middle ground between speed and accuracy. We found that even though surrogate models are indeed faster, they can lack accuracy, especially in cases where bathymetry plays a less controlling role. The super-resolution approach mitigates these errors due to its input that stems from an actual physical computation and thus reduces the prediction to an adjustment. Furthermore, while super-resolution and surrogate models share some overlapping applications, they are not fully identical. Super-resolution could be used to increase the resolution of wave maps or databases, where no spectral information is available. Indeed, if trained on wave maps of an external provider, the prediction time and thus the speed-up is similar to the surrogate model, but with better accuracy than the latter.

Overall, we found that super-resolution has the potential to considerably reduce computation time for forecasts. As seen in subsection 3.3, in certain cases the initial computational costs of the method are quite high. Especially for shorter hindcasts those costs have to be considered. Taking the previously presented times, we can roughly estimate at which hindcast length x_{years}, the lower computation times compensate the upfront calculations. On one hand of the equation, we have x_{years} times the duration of a HR SWAN computation of one year T_{HR}. On the other hand, we have the sum of the training time for one year T_{train}, the computation of one year of HR results T_{HR}, $x_{\text{years}} + 1$ times the LR SWAN computation T_{LR} (the $+1$ is due to the creation of the data set), and finally x_{years} times the conversion time of the neural network T_{NN}. Thus, our estimate for how long a hindcast should be to balance the initial costs is:

$$
\begin{aligned}
x_{\text{years}} T_{\text{HR}} &= T_{\text{HR}} + T_{\text{LR}} + T_{\text{train}} + x_{\text{years}}(T_{\text{LR}} + T_{\text{NN}}) \\
\Leftrightarrow x_{\text{years}} &= \frac{T_{\text{HR}} + T_{\text{LR}} + T_{\text{train}}}{T_{\text{HR}} - T_{\text{LR}} - T_{\text{NN}}} \approx 4\,(\text{years})\,,
\end{aligned}
$$

(with $T_{\text{HR}} = 3.75\,\text{h}$, $T_{\text{LR}} = 4\,\text{min}$, $T_{\text{train}} = 9\,\text{h}$, and $T_{\text{NN}} = 15\,\text{s}$). Note that the values given here are normalized to a 1 year duration, i.e., 1 year of HR results, 1 year of conversion, etc., whereas most of the values in subsection 3.3 refer to our total study time of 2 years (that is, 2 years of HR results, 2 years of conversion). In summary, if the super-resolution approach is applied to a hindcast computation, the desired duration should be at least 4 years long to compensate the upfront computation time, but is especially interesting for 10 years and more, where the gain in computation time is

already more than double. Naturally, this estimate is only a rough indicator and is based on our computations. Different hardware, other grid sizes, and other wave models and regions, all influence the estimate for any given study. While this shows that super-resolution is a viable approach for long-term hindcasts, we still see its main application in operational forecasts, particularly in ensemble forecasts or other situations where some accuracy might be sacrificed for faster outputs. Indeed, we showed that after training the neural network, predictions can be obtained 50-times faster with good accuracy with reference to a standard high-resolution SWAN computation. The errors are often negligible in comparison to the accuracy of spectral wave models and lower than for surrogate models. Commonly, NRMSE of SWAN computations, when compared with buoy measurements, are on the order of 10 % to 20 % for significant wave height, mean wave period, and mean wave direction (see, e.g. Akpınar et al. (2012); Gonçalves et al. (2018); Delpey et al. (2021)). As seen in Fig. 7, in our predictions neither T_{m02} nor Dir have a NRMSE larger than 1.5 %. Even for the significant wave height the error stays below 2 % for the majority of the reconstructed data.

Apart from that, also in cases of larger errors the neural network improves upon the LR input. It implies that it still is more informative and accurate than a LR computation, even when in some situations the predictions might not replace a direct HR computation. Furthermore, the speed and accuracy presented here are likely lower bound estimates and could be improved with a more elaborate pre-processing routine, a refined model architecture and careful training data sampling. Additionally, different loss equations and other neural network architectures like GANs could boost the results substantially, with a potential loss of training / prediction speed and of facility of implementation.

Also, as seen in subsection 3.1, the largest errors at extreme wave heights can be mitigated by adding data from stationary runs to the data set. This is a particularly exciting result, since usually the data of most interest are exactly these extreme cases. By only adding approximately 10 % of additional data, we managed to reduce the mean difference in certain locations by half. While the error slightly increased for lower values of H_s, in practice a difference of around 1 cm is negligible and much smaller than the resolution of the spectral wave model itself.

For a more thorough data selection, existing hindcast databases could be used to train the model extensively by constructing a large and well-sampled data set that covers many possible wave conditions and that would likely improve the results considerably. Here, we focused on a simple approach with only one year of training data, to demonstrate that, even with relatively small training sample sizes, we achieve a robust prediction for various wave conditions.

However, there are some disadvantages of our approach, and of using data-dependent algorithms in general. Here, we trained one model for each wave variable, which increases the training time considerably. Direct HR SWAN computations can output multiple variables in one go, with only a minor effect on the overall run time. Nevertheless, for the neural networks, running the prediction for the three variables affects the overall computation time only slightly and training is a one-time cost. Training one model to predict all variables is feasible, however, it deteriorates the results as commonly reported in the literature (Schultz et al., 2021).

Another possible drawback stems from our implementation of data augmentation. Wave height, period, and direction are iterated over in self-defined ranges, sometimes resulting in combinations of parameters that produce computations that are out of distribution at best and non-physical at worst. We found that if the ranges are chosen inaptly, there is a good chance that the extra data will degrade the overall performance.

Future research can explore various directions of improving the super-resolution approach. For instance, in spite of the fact that convolutional neural networks have the advantage to work on an arbitrary input size, preliminary results show that the trained models are location-specific, even if bathymetry is added as an input, so that for now, a new model has to be trained for each new region. However, thanks to the flexibility regarding the input size, the neural network could be trained on multiple locations at the same time, effectively reducing the number of neural networks needed. While a deeper inspection of this question is out of the scope of this paper, it might be an interesting lead for future investigation.

More generally, given that our approach is data-dependent, the performance of the neural network might degrade rapidly in regions or situations where the wave climate and the corresponding distribution of sea state variables change over small time scales. In such cases, other methods might be more suitable. Nonetheless, the re-calibration time of the model can be reduced considerably with transfer learning (Pan and Yang, 2010).

Another recent approach builds on incorporation of physical equations into the training process to produce so-called Physics-Informed Neural Networks (PINNs) (Gao et al., 2021). Commonly, this is done by adding particular terms, like the constraint of zero divergence for incompressible fluids in the loss equations (Raissi et al., 2019). This forces the neural network to not only produce more physically plausible results, but also helps it generalize better. Recent articles also implemented neural networks directly in a partial differential equations solver, improving accuracy at a lower computational cost than a direct numerical solution (Um et al., 2020; De Avila Belbute-Peres et al., 2020). For a thorough overview of PINNs, see for example Willard et al. (2020). While similar approaches appear difficult to implement in a spectral wave model like SWAN, it might be suitable for other wave models relying on less complex governing equations.

Another outstanding question is how the model performs in terms of wind versus swell seas, which will be looked more deeply into in future research.

The study area exhibits strong bathymetric features, which highly influence the sea state, leading to clearly identifiable wave refraction zones that are relatively easy for the neural network to pick up. Other study areas with less clear features might be more difficult to reconstruct and results could be less coherent in time. Furthermore, the range of angles of the incoming waves is relatively narrow in the study area, leading to a limited number of directional values, which facilitates the model's task. Particularly in the open ocean, this restriction is not given and the model's

performance might be worse or would have to be trained much more extensively to accurately reconstruct all possible temporal and spatial variations.

Lastly, we considered only rectangular, uniform grids for the sake of the simplicity for finding and computing high- and low-resolution pairs and its straightforward implementation into neural networks. Given that many studies use more complex grid structures, more research has to be undertaken to generalize this approach to any type of grid. One promising approach applied in recent fluid mechanics articles is to employ graph neural networks (Sanchez-Gonzalez et al., 2020; Chen et al., 2021; Pfaff et al., 2021). These are based on vertices and edges and thus lend themselves naturally to work with unstructured data, including the irregular meshes commonly found in spectral wave models (Delpey et al., 2021).

5. Conclusions

This study presents a super-resolution method based on a Downsampled Skip-Connection / Multi-Scale neural network. With a total of two years of a high- and low-resolution SWAN computation, we were able to train and evaluate the ability of the network to reconstruct high-resolution results from a coarse input and compare it to the usually proposed surrogate models.

More specifically, the neural network:

- Performed overall very well on the whole year of test data. Even with the worst training run it had a mean absolute error of 0.7 cm, 0.04 s, and 0.3°, for significant wave height, mean wave period, and mean wave direction, respectively. Those errors are negligible in comparison to the resolution of spectral wave models.
- Outperformed the accuracy of surrogate models, while still being 50-times faster than a direct high-resolution SWAN computation, after an initial one-time training cost.
- Showed limitations in ranges with small amounts of training data. This is particularly apparent in the case of significant wave height where the difference to the ground truth can be in some instances more than 50 cm. However, errors can be mitigated considerably with targeted data augmentation.

We are convinced that this approach will be a valuable tool in the future in coastal wave modeling. It complements the toolbox by providing an important alternative to surrogate models, offering a balance between speed and accuracy. It can be used for locally "zoomed-in" global wave models or, like surrogate models, for reducing drastically the computational demands of ensemble forecasting.

6. Open Research

6.1. Data Availability

The SWAN, pre-processing and training scripts along with Jupyter notebooks for the figures are available through GitHub under https://github.com/janfer95/SR_on_SWAN (Kuehn et al., 2022).

CRediT authorship contribution statement

J. Kuehn: Conceptualization, Methodology, Software, Writing - Original Draft. **S. Abadie:** Conceptualization, Writing - Review & Editing, Supervision, Funding acquisition. **B. Liquet:** Validation, Writing - Review & Editing. **V. Roeber:** Conceptualization, Writing - Review & Editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Acknowledgments

This research was carried out in the framework of the E2S-UPPA chair HPC-Waves. The authors acknowledge financial support from E2S-UPPA, the Communauté d'Agglomération Pays Basque (CAPB), and the Communauté Région Nouvelle Aquitaine (CRNA) for the chair HPC-Waves.

References

Akpınar, A., van Vledder, G.P., Kömürcü, M.İ., Özger, M., 2012. Evaluation of the numerical wave model (SWAN) for wave simulation in the Black Sea. Continental Shelf Research 50–51, 80–99. doi:10.1016/j.csr.2012.09.012.

Aloysius, N., Geetha, M., 2017. A review on deep convolutional neural networks, in: 2017 International Conference on Communication and Signal Processing (ICCSP), IEEE, Chennai. pp. 0588–0592. doi:10.1109/ICCSP.2017.8286426.

Bai, G., Wang, Z., Zhu, X., Feng, Y., 2022. Development of a 2-D deep learning regional wave field forecast model based on convolutional neural network and the application in South China Sea. Applied Ocean Research 118, 103012. doi:10.1016/j.apor.2021.103012.

Bajo, M., Umgiesser, G., 2010. Storm surge forecast through a combination of dynamic and neural network models. Ocean Modelling 33, 1–9. URL: https://www.sciencedirect.com/science/article/pii/S1463500309002406, doi:10.1016/j.ocemod.2009.12.007.

Battjes, J.A., Stive, M.J.F., 1985. Calibration and verification of a dissipation model for random breaking waves. Journal of Geophysical Research 90, 9159. doi:10.1029/JC090iC05p09159.

Booij, N., Ris, R.C., Holthuijsen, L.H., 1999. A third-generation wave model for coastal regions: 1. Model description and validation. Journal of Geophysical Research: Oceans 104, 7649–7666. doi:10.1029/98JC02622.

Boudière, E., Maisondieu, C., Ardhuin, F., Accensi, M., Pineau-Guillou, L., Lepesqueur, J., 2013. A suitable metocean hindcast database for the design of Marine energy converters. International Journal of Marine Energy 3–4, e40–e52. doi:10.1016/j.ijome.2013.11.010.

Callens, A., Morichon, D., Abadie, S., Delpey, M., Liquet, B., 2020. Using Random forest and Gradient boosting trees to improve wave forecast at a specific location. Applied Ocean Research 104, 102339. doi:10.1016/j.apor.2020.102339.

Camus, P., Mendez, F.J., Medina, R., 2011. A hybrid efficient method to downscale wave climate to coastal areas. Coastal Engineering 58, 851–862. doi:10.1016/j.coastaleng.2011.05.007.

Chen, J., Hachem, E., Viquerat, J., 2021. Graph neural networks for laminar flow prediction around random two-dimensional shapes. Physics of Fluids 33, 123607. doi:10.1063/5.0064108.

De Avila Belbute-Peres, F., Economon, T., Kolter, Z., 2020. Combining differentiable PDE solvers and graph neural networks for fluid flow prediction. Proceedings of the 37th International Conference on Machine Learning 119, 2402–2411. URL: https://proceedings.mlr.press/v119/de-avila-belbute-peres20a.html.

Delpey, M., Lastiri, X., Abadie, S., Roeber, V., Maron, P., Liria, P., Mader, J., 2021. Characterization of the wave resource variability in the French Basque coastal area based on a high-resolution hindcast. Renewable Energy 178, 79–95. doi:10.1016/j.renene.2021.05.167.

Deng, L., 2012. The mnist database of handwritten digit images for machine learning research. IEEE Signal Processing Magazine 29, 141–142.

Du, X., Qu, X., He, Y., Guo, D., 2018. Single image super-resolution based on multi-scale competitive convolutional neural network. Sensors 18, 789.

Ducournau, A., Fablet, R., 2016. Deep learning for ocean remote sensing: An application of convolutional neural networks for super-resolution on satellite-derived SST data, in: 2016 9th IAPR Workshop on Pattern Recogniton in Remote Sensing (PRRS), IEEE, Cancun, Mexico. pp. 1–6. doi:10.1109/PRRS.2016.7867019.

Eaton, J.W., Bateman, D., Hauberg, S., Wehbring, R., 2020. GNU Octave Version 5.2.0 Manual: A High-Level Interactive Language for Numerical Computations.

Erikson, L., Hegermiller, C., Barnard, P., Ruggiero, P., van Ormondt, M., 2015. Projected wave conditions in the Eastern North Pacific under the influence of two CMIP5 climate scenarios. Ocean Modelling 96, 171–185. doi:10.1016/j.ocemod.2015.07.004.

Fukami, K., Fukagata, K., Taira, K., 2019. Super-resolution reconstruction of turbulent flows with machine learning. Journal of Fluid Mechanics 870, 106–120. doi:10.1017/jfm.2019.238, arXiv:1811.11328.

Gao, H., Sun, L., Wang, J.X., 2021. Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels. Physics of Fluids 33, 073603. doi:10.1063/5.0054312.

Gonçalves, M., Martinho, P., Guedes Soares, C., 2018. A 33-year hindcast on wave energy assessment in the western French coast. Energy 165, 790–801. doi:10.1016/j.energy.2018.10.002.

Gopinath, D., Dwarakish, G., 2015. Wave Prediction Using Neural Networks at New Mangalore Port along West Coast of India. Aquatic Procedia 4, 143–150. doi:10.1016/j.aqpro.2015.02.020.

Guo, Y., Liu, Y., Oerlemans, A., Lao, S., Wu, S., Lew, M.S., 2016. Deep learning for visual understanding: A review. Neurocomputing 187, 27–48. doi:10.1016/j.neucom.2015.09.116.

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778.

Hegermiller, C.A., Antolinez, J.A.A., Rueda, A., Camus, P., Perez, J., Erikson, L.H., Barnard, P.L., Mendez, F.J., 2017. A Multimodal Wave Spectrum–Based Approach for Statistical Downscaling of Local Wave Climate. Journal of Physical Oceanography 47, 375–386. doi:10.1175/JPO-D-16-0191.1.

Huang, L., Jing, Y., Chen, H., Zhang, L., Liu, Y., 2022. A regional wind wave prediction surrogate model based on CNN deep learning network. Applied Ocean Research 126, 103287. doi:10.1016/j.apor.2022.103287.

James, S.C., Zhang, Y., O'Donncha, F., 2018. A machine learning framework to forecast wave conditions. Coastal Engineering 137, 1–10. doi:10.1016/j.coastaleng.2018.03.004, arXiv:1709.08725.

Jörges, C., Berkenbrink, C., Gottschalk, H., Stumpe, B., 2023. Spatial ocean wave height prediction with CNN mixed-data deep neural networks using random field simulated bathymetry. Ocean Engineering 271, 113699. doi:10.1016/j.oceaneng.2023.113699.

Kim, B., Azevedo, V.C., Thuerey, N., Kim, T., Gross, M., Solenthaler, B., 2019. Deep Fluids: A Generative Network for Parameterized Fluid Simulations. Computer Graphics Forum 38, 59–70. doi:10.1111/cgf.13619.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization, in: Bengio, Y., LeCun, Y. (Eds.), 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Kuehn, J., Abadie, S., Liquet, B., Roeber, V., 2022. Janfer95/SR_on_SWAN: SR_on_SWAN. Zenodo. doi:10.5281/zenodo.7108005.

Lloyd, D.T., Abela, A., Farrugia, R.A., Galea, A., Valentino, G., 2022. Optically Enhanced Super-Resolution of Sea Surface Temperature Using Deep Learning. IEEE Transactions on Geoscience and Remote Sensing 60, 1–14. doi:10.1109/TGRS.2021.3094117.

Londhe, S.N., Shah, S., Dixit, P.R., Nair, T.M., Sirisha, P., Jain, R., 2016. A coupled numerical and artificial neural network model for improving location specific wave forecast. Applied Ocean Research 59, 483–491. doi:10.1016/j.apor.2016.07.004.

Lucero, F., Stringari, C.E., Filipot, J.F., 2023. Improving WAVEWATCH III hindcasts with machine learning. Coastal Engineering 185, 104381. doi:10.1016/j.coastaleng.2023.104381.

Madsen, O.S., Poon, Y.K., Graber, H.C., 1988. SPECTRAL WAVE ATTENUATION BY BOTTOM FRICTION: THEORY. Coastal Engineering Proceedings 1, 34. doi:10.9753/icce.v21.34.

Mahdavi-Meymand, A., Sulisz, W., 2023. Application of nested artificial neural network for the prediction of significant wave height. Renewable Energy 209, 157–168. doi:10.1016/j.renene.2023.03.118.

Ngiam, J., Chen, Z., Chia, D., Koh, P., Le, Q., Ng, A., 2010. Tiled convolutional neural networks. Advances in neural information processing systems 23.

O'Donncha, F., Zhang, Y., Chen, B., James, S.C., 2018. An integrated framework that combines machine learning and numerical models to improve wave-condition forecasts. Journal of Marine Systems 186, 29–36. doi:10.1016/j.jmarsys.2018.05.006.

Pan, S.J., Yang, Q., 2010. A Survey on Transfer Learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359. doi:10.1109/TKDE.2009.191.

Perez, L., Wang, J., 2017. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. arXiv:1712.04621 [cs] arXiv:1712.04621.

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., Battaglia, P.W., 2021. Learning Mesh-Based Simulation with Graph Networks, in: 9th International Conference on Learning Representations, ICLR 2021, arXiv. arXiv:2010.03409.

Pinault, J., Morichon, D., Delpey, M., Roeber, V., 2022. Field observations and numerical modeling of swash motions at an engineered embayed beach under moderate to energetic conditions. Estuarine, Coastal and Shelf Science 279, 108143.

Prechelt, L., 1998. Early stopping-but when?, in: Neural Networks: Tricks of the Trade. Springer, pp. 55–69.

Raissi, M., Perdikaris, P., Karniadakis, G., 2019. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. Journal of Computational Physics 378, 686–707. doi:10.1016/j.jcp.2018.10.045.

Rawat, W., Wang, Z., 2017. Deep Convolutional Neural Networks for Image Classification: A Comprehensive Review. Neural Computation 29, 2352–2449. doi:10.1162/neco_a_00990.

Sanchez-Gonzalez, A., Godwin, J., Pfaff, T., Ying, R., Leskovec, J., Battaglia, P.W., 2020. Learning to Simulate Complex Physics with Graph Networks, in: Proceedings of the 37th International Conference on Machine Learning, arXiv. arXiv:2002.09405.

Schultz, M.G., Betancourt, C., Gong, B., Kleinert, F., Langguth, M., Leufen, L.H., Mozaffari, A., Stadtler, S., 2021. Can deep learning beat numerical weather prediction? Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 379, 20200097. doi:10.1098/rsta.2020.0097.

SHOM-Service Hydrographique Et Océanographique De La Marine, 2015. MNT Bathymétrique de façade Atlantique (Projet Homonim). doi:10.17183/MNT_ATL100M_HOMONIM_WGS84.

Shorten, C., Khoshgoftaar, T.M., 2019. A survey on Image Data Augmentation for Deep Learning. Journal of Big Data 6, 60. doi:10.1186/s40537-019-0197-0.

Sonogashira, M., Shonai, M., Iiyama, M., 2020. High-resolution bathymetry by deep-learning-based image superresolution. PLOS ONE 15. doi:10.1371/journal.pone.0235487.

Stengel, K., Glaws, A., Hettinger, D., King, R.N., 2020. Adversarial super-resolution of climatological wind and solar data. Proceedings of the National Academy of Sciences 117, 16805–16815. doi:10.1073/pnas.1918964117.

Su, H., Wang, A., Zhang, T., Qin, T., Du, X., Yan, X.H., 2021. Super-resolution of subsurface temperature field from remote sensing observations based on machine learning. International Journal of Applied Earth Observation and Geoinformation 102, 102440. doi:10.1016/j.jag.2021.102440.

Thiria, S., Sorror, C., Archambault, T., Charantonis, A., Bereziat, D., Mejia, C., Molines, J.M., Crépon, M., 2023. Downscaling of ocean fields by fusion of heterogeneous observations using Deep Learning algorithms. Ocean Modelling 182, 102174. doi:10.1016/j.ocemod.2023.102174.

Tolman, H.L., 2009. User manual and system documentation of WAVEWATCH III TM version 3.14. Technical note, MMAB Contribution 276, 220.

Um, K., Brand, R., Yun, Fei, Holl, P., Thuerey, N., 2020. Solver-in-the-loop: Learning from differentiable physics to interact with iterative PDE-Solvers. 34th Conference on Neural Information Processing Systems 1, 1–37. arXiv:2007.00016.

Wang, Z., Chen, J., Hoi, S.C.H., 2020. Deep Learning for Image Super-resolution: A Survey, in: IEEE Transactions on Pattern Analysis and Machine Intelligence, arXiv. arXiv:1902.06068.

Willard, J., Jia, X., Xu, S., Steinbach, M., Kumar, V., 2020. Integrating physics-based modeling with machine learning: A survey. arXiv 1, 1–34. arXiv:2003.04919.

A. Appendix

A.1. Implementation of Fully-Connected and Convolutional Neural Networks

Subsection 2.2 shortly explores some less complex alternatives to the DSC/MS model and shows that even though FCNNs have a similar, albeit slightly worse performance, they tend to have a cumbersome amount of parameters, easily reaching 200-300 million, i.e., model sizes of about 3 GB. Another disadvantage that was not discussed in the main text is that FCNNs are inherently not applicable to other study regions, due to their structure. The input of a FCNN needs to always have the exact same size (i.e., here 10x10), whereas CNNs and the DCS/MS model are more flexible in this regard and can accept different input sizes. This is due to the fact that convolutional layers learn the weights of

Location		Bias [cm]	RMSE [cm]	NRMSE [%]
P1	With DA	**-0.4**	**1.7**	**1.0**
	Without DA	-0.5	3.1	1.8
P2	With DA	-0.3	**1.3**	**0.8**
	Without DA	**0.2**	1.5	0.9
P3	With DA	**-0.1**	**2.0**	**1.2**
	Without DA	-1.2	2.6	1.5

Table 3
Same as Table 2, but for the whole time series, i.e., all wave heights.

a kernel, that runs over the input image. FCNNs on the other hand, have weights that are fixed to an exact input node, and are thus only able to work with inputs of the exact same size.

The training procedure of the three neural networks is the same as described in the Workflow (subsection 2.4). This means that all runs used an ADAM optimizer, with early stopping and a patience of 30, and minimized on a mean average error loss function.

We implement the FCNN with an initial flattening layer, converting the size of the input from (10x10) to (100x1). It follows two hidden dense layers of size 3000 and 10 000 and ReLU activation functions, before using a dense layer of size 25 600 without activation function that is reshaped to the same format (160x160) as the high-resolution ground truth. The left flowchart in Fig. 12 summarizes this.

For the convolutional neural network we use an initial nearest-neighbor upsampling layer with a factor of 16, bringing the LR input to (160x160). Consequently, three convolutional layers with 32 filters each, a kernel of 3x3, and padding follow. Each of them is activated by a ReLU function. Lastly, a similar convolutional layer combines the 32 filters into 1, without using any activation function. A summarizing flowchart can be found on the right-hand side of Fig. 12.

We want to emphasize that this comparison does not aim to show the superiority of the DSC/MS model in super-resolution over all possible neural network architectures. Quite on the contrary, we highlight that it is certainly feasible to find more performant architectures, especially with large models like Generative Adversarial Networks or ResNet-50. However, our proposed approach is easy to implement, is adapted to our case, has a small number of parameters and can be run and trained on commodity hardware in a reasonable amount of time, while having an accuracy that is more than enough for general wave model use cases (see the discussion in section 4 for a more in-depth analysis of this point).

A.2. Tables

A.3. Additional Figures

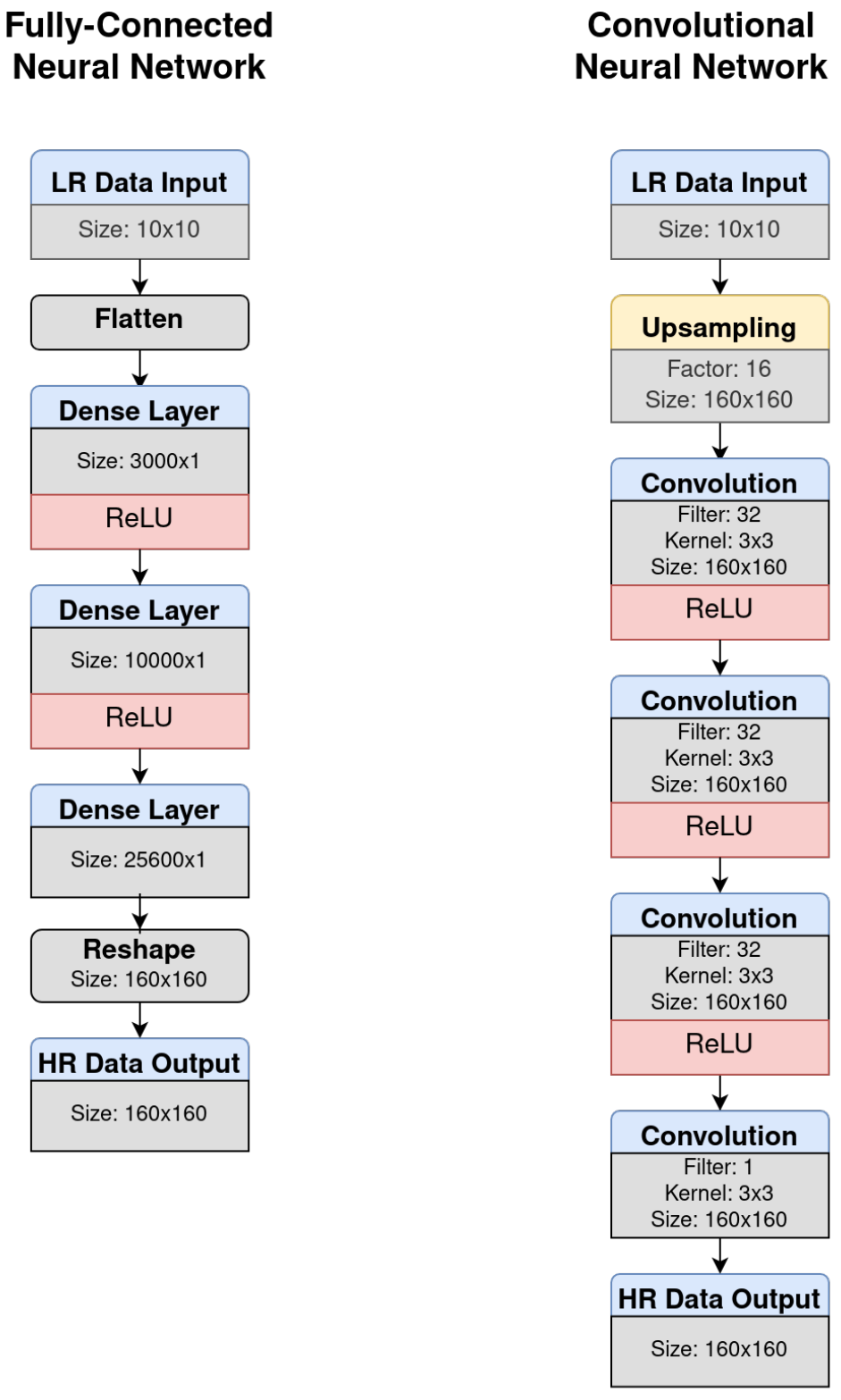

Figure 12: Flowchart for the neural network architectures of the Fully-Connected Neural Network and the Convolutional Neural Network that were compared to the DSC/MS model in subsection 2.2.

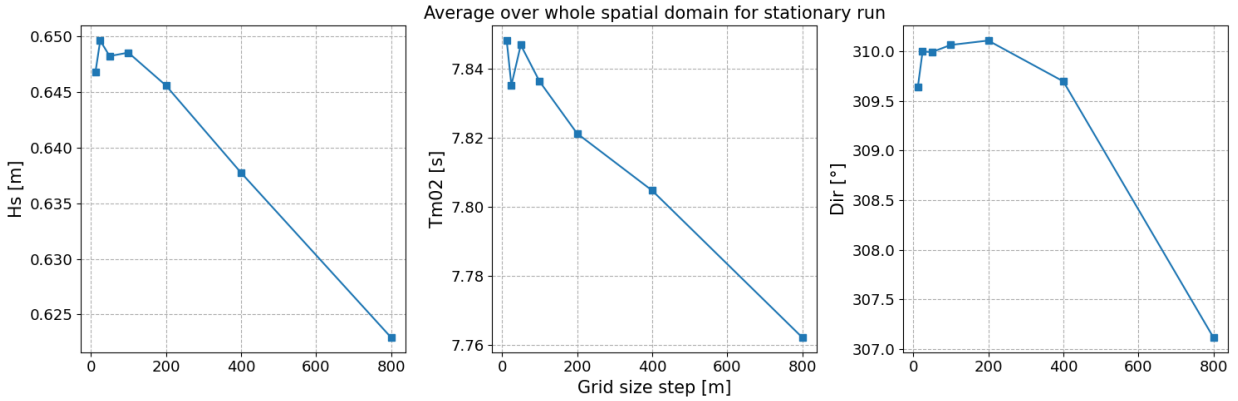

Figure 13: Simple convergence analysis of stationary SWAN runs for different grid size steps. For each grid size step, the mean value over the whole spatial domain for a stationary run is computed. Differences between $25\,\mathrm{m}$ and $50\,\mathrm{m}$ grid size steps are negligible for all three variables, being on the order of less than $0.5\,\%$

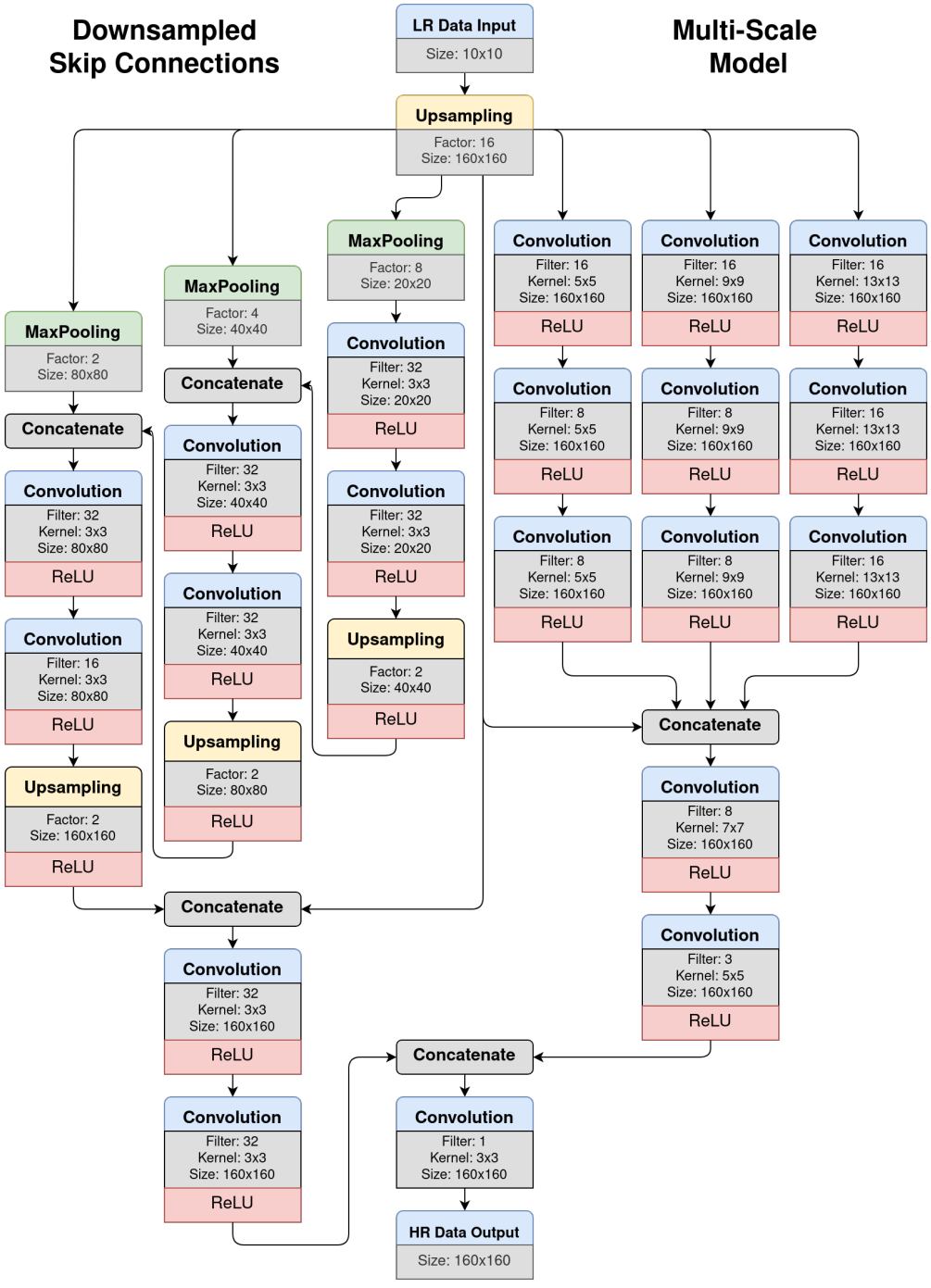

Figure 14: Flowchart of the neural network architecture of the DSC/MS model used in this study.

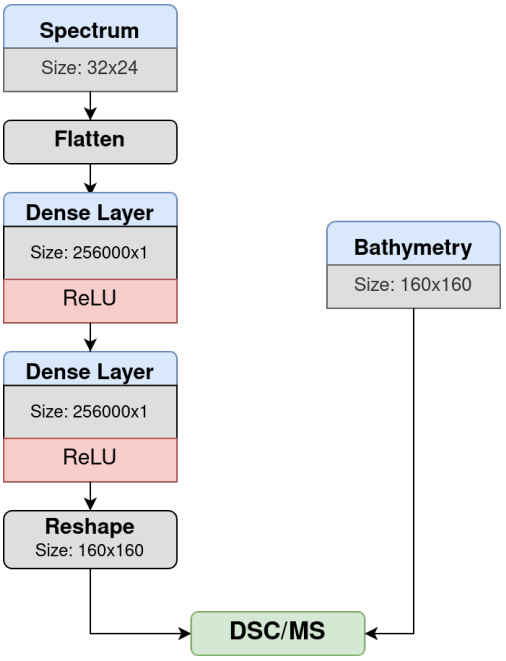

Figure 15: The network architecture of the surrogate model used in this study. The spectrum is transformed by dense layers and is passed together with the bathymetry to the DSC/MS model.

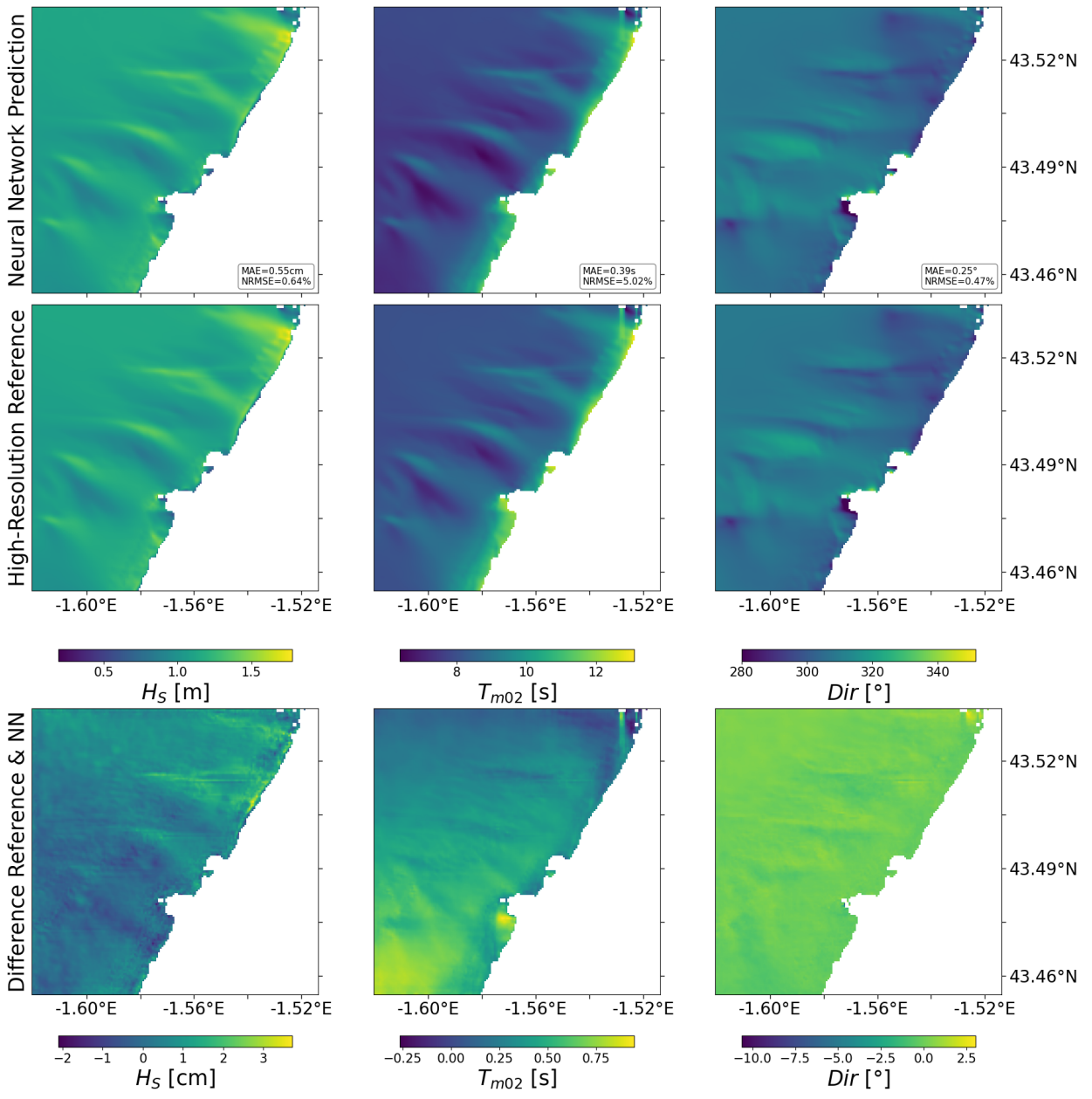

Figure 16: The same Figure as Fig. 4 in the main text, but for the surrogate model. Note that the low-resolution input is missing, since for this neural network the input is a spectrum and the bathymetry.

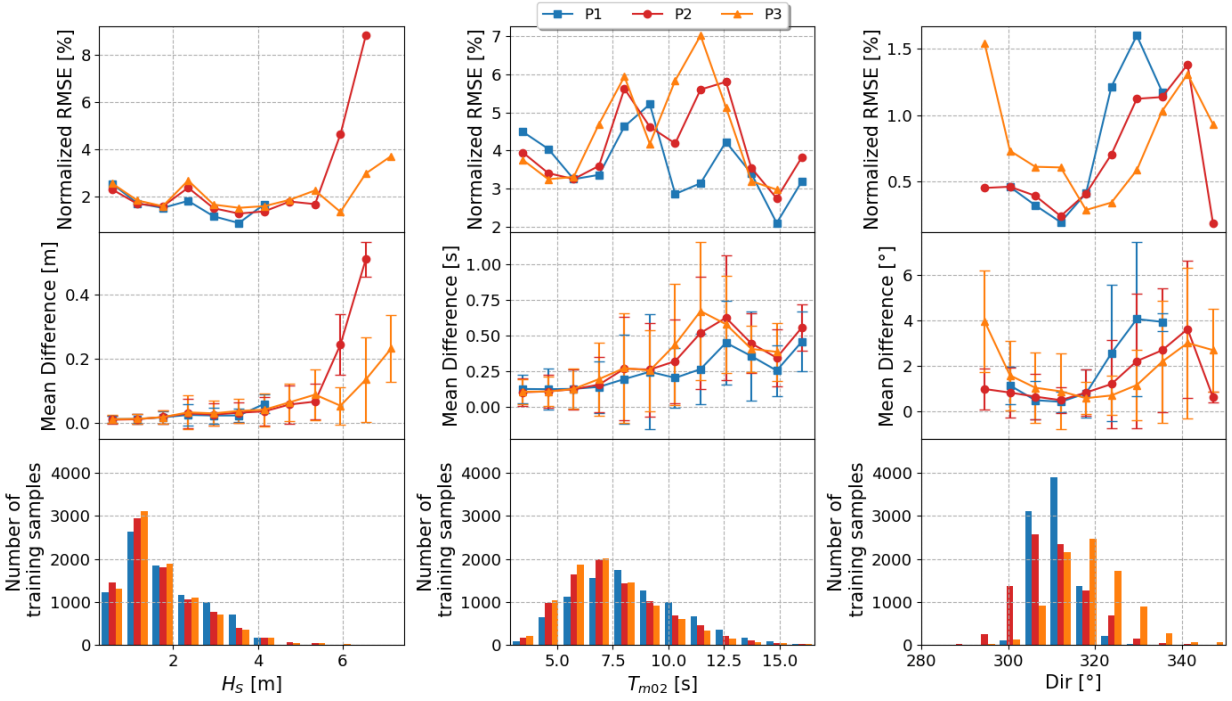

Figure 17: The same Figure as Fig. 7 in the main text, but for the surrogate model.